HERSCHEL
HIFI ICU On-Board Software
IFSI Test Report
Document Ref.: IFSI/OBS/TR/2006-001

Issue: 1.12

Prepared by: Anna Maria Di Giorgio

Document Status Sheet:

Document Title: Herschel HIFI ICU OBS Software IFSI Test Report		
Issue	**Date**	**Reason for Change**
Issue 1.1	20/05/2006	OBS 3.6 tests on AVM1@IFSI
Issue 1.2	20/06/2006	OBS 3.6 tests on PFM @CGS
Issue 1.3	17/07/2006	OBS 4.0 tests on AVM1@IFSI
Issue 1.4	01/09/2006	OBS 4.1 tests on AVM1@IFSI
Issue 1.5	01/09/2007	OBS 5.2 tests on AVM1@IFSI
Issue 1.6		OBS 5.3 tests on AVM1@IFSI
Issue 1.7	07/12/2007	OBS5.4 tests on AVM1@IFSI
Issue 1.8	19/07/2008	OBS5.7/5.8 tests on AVM1@IFSI
Issue 1.9	04/08/2008	OBS5.8.1 tests on AVM1@IFSI
Issue 1.10	17/01/2009	OBS5.9 tests on AVM1@IFSI
Issue 1.11	17/02/2009	OBS6.0/6.1 tests on AVM1@IFSI
Issue 1.12	24/02/2009	OBS6.1.1 tests on AVM1@IFSI

Document Change record:

1 Introduction

1.1 Purpose of the document

This document reports the result of the tests executed at IFSI with OBS 6.1.1 onboard ICU AVM1 model. It contains a copy of the test procedures document (AD17) and corresponds to the test cases described in the SVVP AD1.
OBS 6.1.1 contains a quick fix to the problem highlighted in SPR 2216, implying only a modification on a table initialized at startup. This modification has impact only on the new HIFI_LOU_T_check_on Telecommand and no other functionality onboard is affected by it. Therefore, OBS 6.1.1 has been tested only for the LCU temperature limit check functionality with a new dedicated test procedure described in section 4.3 (TP3).

1.2 Acronyms and Glossary

BC	Bus Controller
CDMS	Command and Data Management System
DM	Data Memory (DSP)
DTST	Dedicated Test Software Tools
EGSE	Electrical Ground Support Equipment
ESA	European Space Agency
HERSCHEL	Herschel Space Observatory
HK	Housekeeping
NA	Not Applicable
PFM	Protoflight Model
S/S	Subsystem
SUT	Software Under Test
TBC	To Be Confirmed
TBD	To Be Defined
TBW	To Be Written
TC	Telecommand
TM	Telemetry

1.3 Document List

1.3.1 Applicable Documents

Document Reference	Name	Number/version/date
AD1	HIFI ICU OBS SVVP	IFSI/OBS/PL/2005-001 Issue 3.0 – 11/10/2005
AD2	HIFI ICU OBS Software Specifications Document	IFSI/OBS/SP/2002-001 Issue 2.1 – 15/04/2005
AD3	Herschel DPU/ICU OBS Product Assurance Plan	IFSI/OBS/PL/2000-001 Issue 1.1 –02/04/2001
AD4	Packet Structure Interface Control Document	SCI-PT-ICD-7527 Issue 6.0 – 25/01/2008
AD5	Herschel/Planck Instrument Data Rates	H-P-1-ASPI-TN-0204 Issue 1.0 – 15/01/2002
AD6	DPU Switch-on procedure	CNR.IFSI..2001.TR01 Issue 1.0 – 12/10/2001

AD7	Software – Part 1: principles and requirements	ECSS E-40 part1 Rev. B 24/05/2002
AD8	HIFI TC Packet ICD	SRON-U/HIFI/SP/2001-1 Issue 1.8
AD9	HIFI TM Packet ICD	SRON-U/HIFI/SP/2001-2 Issue 1.8
AD10	HIFI HK Packet ICD	SRON-U/HIFI/SP/2001-3 Issue 1.12
AD11	HIFI Command Specification	SRON-U/HIFI-SP-2001-4 Issue 1.9
AD12	HIFI Internal Databusses ICD	SRON-U/HIFI-SP-2001-10 Issue 1.5
AD13	HIFI OBS Test Procedures	SRON-U/HIFI/PR/2005-003
AD14	DPU/ICU OBS Development Plan	CNR IFSI-2004-PL-001 Issue 1 29/10/2004
AD15	HIFI OBS WBS Test procedure	IFSI/OBS/SP/2005-002 Issue1.0 05/10/2005
AD16	HIFI OBS VM Test Procedure	IFSI/OBS/SP/2005-001 Issue1.0 05/10/2005
AD17	HIFI ICU OBS Test Procedure	IFSI/OBS/TP/2006-001 Issue1.0 10/05/2006
AD18	HIFI IFSI Test Procedures	Issue 4.4
AD19	HIFI OBS Software Release Notice Issue 6.0	SRON_U_HIFI_TN_2003_009 Issue 6.0 16/02/2009
AD20	HIFI OBS Software Release Notice Issue 6.1	SRON_U_HIFI_TN_2003_009 Issue 6.1 17/02/2009

1.3.2 Reference Documents

Document Reference	Name	Number/version
RD1	Guide to applying the ESA software engineering standards to small software projects	BSSC(96)2 Issue 1 – May 1996
RD2	HIFI Packet Logger User Manual	SRON_U_HIFI_TN_2005_6 issue 1.0

2 Validation Tests

2.1 Features to be tested

The following set of features will be tested (each item in the list may be composed of a set of sub-items):

1. ICU switch-on procedure implementation: from the power-on PROM boot up to the switch on of all the subsytems.
2. Telecommand Verification Service implementation, i.e. TC ingestion, verification and generation of the relative acceptance report
3. Housekeeping & Diagnostic Data Reporting Service implementation, i.e. acquisition and packing of both ICU internal and S/S HK parameters
4. Memory Management Service implementation: identification and execution of ICU commands,
5. Function Management Service implementation: identification and execution of ICU commands, identification and transmission of commands to the S/Ss both directly and by VM activation
6. Test Service implementation: connection test
7. Science Data Transfer Service implementation: reception of Science packets from spectrometers and relative packing
8. Time management Service implementation
9. Packet Transmission Control

2.2 Test Deliverables

At the end of the acceptance tests, the following items will be delivered:

1. Test reports;
2. All input files used during tests;
3. Test log files

2.3 Test environment

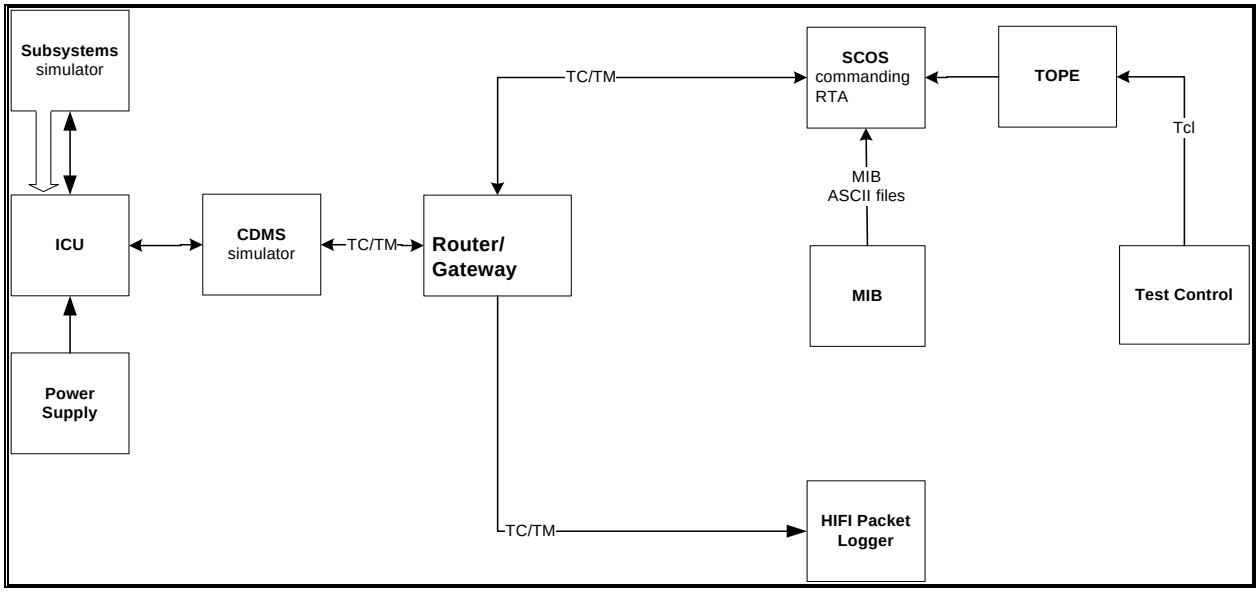

Figure 1 OBS Test environment

The OBS validation activities will be carried out in the test environment described in Figure 1.
The following equipment must be installed at IFSI in order for the tests to be carried out:

1. SRON provided S/S simulator (HW and SW), installed on a dedicated PC
2. SCOS2000 system, consisting of:
    a. Personal computer running Linux SUSE 7.3
    b. SCOS2000 Version 2.3e + patch 5.1
    c. HIFI MIB
3. Router (java application sw provided by SRON)
4. CDMS simulator, provided by RAL, consisting of:
    i. Personal Computer running Windows
    ii. DDC 1553 interface board version BU-65549
    iii. Application SW CDMS_SIM version 2.4, including the buslists and the telecommands that will be used in the tests.
5. ICU external power supply
6. Dedicated SW tools to support the tests:
    a. ObswLoader script resident on the SCOS2000 computer, used to uplink the series of TC (6,2) commands with the image of the OBS executable. Loading procedure is described in RD3.
    b. TCGEN application SW version1 to generate the list of telecommands to uplink the OBS.
    c. HIFI_packet_Logger. It connects to the SCOS Router and so it can be run on any machine connected to the network.
    d. Tools for analysing the TM data provided by the ICU not directly analysed by SCOS2000.
    e. CRC program to compute the CRC checkword from a series of data words. It will be resident on any machine.
7. HW test equipment

The HIFI OBS will be available on the SCOS2000 system as a set of TC (6,2) telecommands produced with the CGS provided TCGEN procedure.
The CDMS Simulator will run the CDMS_SIM software package version 2.5.
The following buslists will be available: HIFI_Nominal.txt and a copy of it with the TIME_SYNC directive removed (i.e. the CDMS will not send the 1s periodic time sync information as specified in AD4.

2.4 Test case pass/fail criteria

Test criteria are based on the inspection of the provided TM packets and of the log files of the Subsystem simulator. For each test a reference result shall be indicated.
A test is passed when all the mandatory functions to be verified with the test are checked successfully by comparing the test result with the expected result.

3 Test Specification

3.1 Test preparation

The following steps should be performed to setup the system ready for use:

AD1 Start the SRON S/S Simulator:
    a. switch on the PC hosting the simulator board.
    b. login
    c. go to the "/home/..subsyssim/subss/scripts" directory
    d. launch the simulator starting script. The script to be used for every test case will be specified in the related test procedure.

AD2 Start SCOS2000
    a. Login as user
    b. Type startx and press return
    c. Open a terminal window.
    d. Change directory to local-bin (type cd local-bin)
    e. Execute script StartRouterHIFI (type ./StartRouterHIFI)
    f. Go to /home/sops23e
    g. Execute script set_links_HIFI_MIB to use the MIB version specified in the test procedure

h. Import the HIFI MIB via the command IMPT
i. Execute script s2.start
   i. Select EGSEServ, MON1 and MSTK2 buttons, EXIF
   ii. press Start and confirm
j. When desktop appears, login as username Matt (passwd Matt) with role SUPE_001
k. Disable warning bell. Press button Alarm tone disable, click alarm checkbox and confirm.
l. On the MON1 (Telemetry Desktop) window, select AND button at bottom left of the window and choose ICU _Housekeeping; select List button
m. The procedure is completed when all the selected windows/services appear.

At this point the SCOS2000 system is up and running

AD3 **Start the CDMS Simulator. On the PC hosting the CDMS simulator:**

a. Click on icon CDMS_SIM
b. On the "Select Buslist" button, select the HIFI_Nominal (27TMslots/sec) buslist
c. Click on Launch Router Command Interface and specify the address of the SCOS2000 /router machine
d. Click on Connect
e. On the "Select Command to send" option select NAME_CLIENT; write "hifi" and click Send Command
f. On the "Select Command to send" option select ADD_CLIENT, write "400" and click Send Command
g. Click on Close Without Sending
h. Click on Start/Stop BC button

The CDMS is ready to be started

AD4 **Start the HIFI_packet_logger tool**

3.2 Test Initial Data

Experimenters	Anna Maria Di Giorgio (IFSI) Lorenzo Piazzo (INFOCOM)
Date	24/02/2009
Location	IFSI Rome
ICU Identification	AVM1
OBS identification	OBS Version 6.1.1
CDMS sim Version/ buslist	CDMS v 2.4 Buslist: HIFI_nominal (27 TM slots)
MIB version	MIB 144
Subsystems simulator	SRON Simulator Version 1.04

4 Test procedures

The procedures described in this section are based on the test cases described in section 5 of AD1.

4.1 TP1: ICU_SWITCH_ON

The ICU switch on procedure is implemented by the Boot Software (BSW), a SW resident in PROM and separated by the OBS.
The BSW has a dedicated Test Plan/Procedures document provided by CGS (refer to TBD applicable document) .
No BSW tests are foreseen in this procedure.

Tests executed on 24/02/2009. CDMS Telemetry log files available at IFSI.

Step	Action	Expected Reaction	Observed Reaction	Notes
1.	Switch-on the ICU. At this point the Boot Software loads the resident OBS image from the EEPROM to PM. After completion, the Boot SW stops.	An TM event (5,1) should be received by CDMS simulator and visible in the "Telemetry Packet Info" window.	**OK**	
2.	On MSTK2 Send HIFI_Force_boot to start the OBS from partition 1.	On the alphanumeric Displays of MON1 window (SCOS2000), the ICU Housekeeping shall be displayed.	**OK**	External reset button recycling necessary.
3.	Verify Voltage and Current	See A1.2	Not done	
4.	Check ICU HK contents	See A1.1	Not done	
5.	Note OBS Version		6.1	
6.	Switch-off the ICU.		OK	
7.	Repeat step 1.	An event TM (5,1) should be received by CDMS.	OK	
8.	Upload the OBS via the EGSE Router. This can either be done in two ways: 1) with OBSM.csh: on a terminal window of the computer hosting SCOS 2000, type the following commands: > cd > cd /DPU_HIFI_TC >../local-bin/ObswLoader -apid 1024 -dpu -interval 250 DmPageTc0*.dm 2) by using a stack-file. Any event TM of type (5,4) shall be reported	An event TM (5,1) should be received by CDMS for each one of the TC ingested during the upload procedure.	OK	No 5,4 events generated
9.	On MSTK2 Send HIFI_Load_boot to start the OBS.	On the alphanumeric Displays of MON1 window (SCOS2000), the ICU Housekeeping shall be displayed.	OK	OBS 6.1.1 version number confirmed
10.	Repeat steps 3, 4, 5.		**OK**	ICU HK cannot be checked on AVM1.
11.	*Save the image in the EEPROM:* On MSTK2 send HIFI_EEPROM_write TC (specify the end address indicated in the OBS Release Note) Wait for completion. (visible when the TCHIST window of SCOS-2000 is open)	A TM (1,1) shall be provided by ICU at the TC acceptance. A TM (1,7) packet shall be provided by ICU at the end of Tc execution. On the TCHIST window of SCOS2000 check the command completion	**OK**	Commanded end address 0x18fff.
12.	Load the OBS resident in the EEPROM: Repeat step 6, 1, 2, 3, 4, 5		**OK**	External reset button recycling necessary.
13.	Start the subsystems simulator testcase01 script.	On the PC hosting the subsystems simulator verify that all 6 simulator processes are correctly started.	**OK**	
14.	On MON1 AND select Essential HK window	Values should be White (invalid)	**OK**	
15.	On MON1 AND select	Values should be white	**OK**	

	HRSH_status window	(invalid)		
16.	On MSTK2 window, send HIFI_Notify_PDU_Status to notify HRS-H switch-on	In MON1 AND ICU_Housekeeping: HI_SUBSYSTEM_S =0x01. HI_HRSH_S=on. The HI_HRSH_HK_S shall become VALID within two acquisition periods.	**OK**	
17.	On MON1 AND select HRSH_status window	Values should turn red or green (valid)	**Not performed**	All subsystems were switched on with a single command and the HK validity was checked globally
18.	On MON1 AND select HRSV_status window	Values should be white (invalid)	**Not performed**	
19.	On MSTK2 window, send HIFI_Notify_PDU_Status to notify HRS-V switch-on (in addition to the previously switched on subsystems).	In MON1 AND ICU_Housekeeping: HI_SUBSYSTEM_S =0x03. HI_HRSV_S=on. The HI_HRSV_HK_S shall become VALID within two acquisition periods.	**Not performed**	
20.	On MON1 AND select HRSVstatus window	Values should turn red or green (valid)	**Not performed**	
21.	On MON1 AND select WBSH_status window	Values should be white (invalid)	**Not performed**	
22.	On MSTK2 window, send HIFI_Notify_PDU_Status to notify WBS-H switch-on (in addition to the previously switched on subsystems).	In MON1 AND ICU_Housekeeping: HI_SUBSYSTEM_S =0x07. HI_WBSH_S=on. The HI_WBSH_HK_S shall become VALID within two acquisition periods.	**Not performed**	
23.	On MON1 AND select WBSH_status window	Values should turn red or green (valid)	**Not performed**	
24.	On MON1 AND select WBSV_status window	Values should be white (invalid)	**Not performed**	
25.	On MSTK2 window, send HIFI_Notify_PDU_Status to notify WBS-V switch-on (in addition to the previously switched on subsystems).	In MON1 AND ICU_Housekeeping: HI_SUBSYSTEM_S = 0x0F. HI_WBSV_S=on. The HI_WBSV_HK_S shall become VALID within two acquisition periods.	**Not performed**	
26.	On MON1 AND select WBSVstatus window	Values should turn red or green (valid)	**Not performed**	
27.	On MON1 AND select LCU_status window	Values should be white (invalid)	**Not performed**	
28.	On MSTK2 window, send HIFI_Notify_PDU_Status to notify LCU switch-on (in addition to the previously switched on subsystems).	In MON1 AND ICU_Housekeeping: HI_SUBSYSTEM_S = 0x1F. HI_LCU_S=on.	**Not performed**	
29.	On MON1 AND select LCU status window	Values should turn red or green (valid)	**Not performed**	
30.	On MON1 AND select FCU_status window	Values should be white (invalid)	**Not performed**	
31.	On MSTK2 window, send HIFI_Notify_PDU_Status to notify FCU switch-on (in addition to the previously switched on subsystems).	In MON1 AND ICU_Housekeeping: HI_SUBSYSTEM_S = 0x3F. HI_FCU_S=on.	**Not performed**	
32.	On subsystems simulator analyse the cmd and hk.log files to check Hk requests timing. See AD13, Annex C.2 for the description of the nominal HK Polling scenario.	Verify that the overall nominal HK request procedure is shorter that 1 sec. Verify that the time interval between two subsequent LCU requests is never shorter than 2msec.	**OK**	
33.	On MON1 AND select FCU status window	Values should turn red or green (valid)	**OK**	
34.	On MON1 AND select Essential	Values should be red or green	**OK**	

	HK window	(valid)		
35.	On SCOS2000, on the MSTK2 window, send HIFI_RESET command.	Verify that the SW reset has been performed: check that the APID counter of the HK packets is restarted and that the ICU HK have been reset to initial startup values.	**OK**	**External reset button recycling necessary**
36.	On SCOS2000, on the MSTK2 window, send H_jump_to_boot command	Verify that the unit reset has been performed and that the Boot Software has been restarted successfully: a (5,1) event packet shall be provided.	**OK**	**External reset button recycling necessary**
37.	Repeat steps 2,3,4,5		**OK**	**External reset button recycling necessary**

4.2 TP2 - COMMAND_ACCEPTANCE

Assuming that the minor patch implemented in OBS 6.1.1 has no impact on the functionalities checked with TP2, noTP2 tests have been performed on OBS 6.1.1.

Step	Action	Expected Reaction	Observed Reaction	Notes
1.	On MSTK2 window, release the HIFI_connection_test TC (17.1)	On the TM log window of the CDMS, verify reception of: TM (1,1), (17,2).	**Not performed**	
2.	On MSTK2 window, release the HIFI_enable_time_verification TC (9.7) to request a time verification report.	On the TM log window of the CDMS, verify reception of: -TM (1,1) and TM (9,9) within 500msec from the TC sending. The time reported in the report shall be equal to the expected value of the onboard time at the next CDMS synch: the number of seconds shall be equal to the number of seconds of the time in the DFH of the report + 1sec. The word containing the fractions of seconds shall be 0.	**Not performed**	
3.	Stop CDMS. Switch off CDMS simulator. Open CDMS file APID2RT.txt and associate HIFI with APID 0x300 (modify the line 0x400 16 HIFI with 0x300 16 HIFI) ; this is needed to force the CDMS to send TCs with wrong APID to HIFI.		**Not performed**	
4.	Repeat steps 1, 2 and 3 of TP1		**Not performed**	
5.	Set CDMS TC source to Local Commands. Send APID_test TC (see A2, Table 1) to test OBS reaction against wrong APID in TC	Click “view log file” button in telemetry packet info. Verify reception of TM (1,2) with failure code 0 in word 11 of the packet and 0x300 in word 12.	**Not performed**	
6.	Stop CDMS. Switch off CDMS simulator Open CDMS file APID2RT.txt and change HIFI’s APID back to nominal.		**Not performed**	
7.	Set CDMS TC source to Local Commands.		**Not performed**	
8.	Send Length_test TC (see A2, Table 1) to test OBS reaction against wrong TC packet length.	Verify reception of TM (1,2) with failure code 1	**Not performed**	
9.	Send CRC_test TC (see A2, Table 1) to test OBS reaction against wrong TC packet checksum.	Verify reception of TM (1,2) with failure code 2	**Not performed**	
10.	Send Type_test TC (see A2, Table 1) to test OBS reaction against wrong TC packet type.	Verify reception of TM (1,2) with failure code 3	**Not performed**	
11.	Send Subtype_test TC (see A2, Table 1) to test OBS reaction against wrong TC packet subtype.	Verify reception of TM (1,2) with failure code 4	**Not performed**	
12.	Send Ack0_test TC (see A2, Table 1) to test OBS reaction against different TC “ack” bits	Verify that only TM (17,2) is received	Not performed	
13.	Send Ack3_test TC (see A2, Table 1) to test OBS reaction against different TC “ack” bits	Verify that only TM (1,1) and TM (17,2) are received	Not performed	
14.	Send Ack5_test TC (see A2, Table 1) to test OBS reaction against different TC “ack” bits	Verify that only TM (1,1) and TM (17,2) are received	Not performed	
15.	Send Ack9_test TC (see A2, Table 1) to test OBS reaction against different TC “ack” bits	Verify that TM (1,1), TM (1,7) and TM (17,2) are received	Not performed	

16.	Set CDMS TC source to Router.			
17.	On MSTK2 window, release HIFI_report_transmission and check that all packets are enabled on Enabled_Packets window of MON1 AND.	Verify reception of TM (1,1) and of HIFI_TM_generation_ status_report (14,4)	**Not performed**	
18.	On MSTK2 window, release HIFI_disable_transmissin to disable the following TM packets: - HIFI_Conn - Time_verif - HIFI_HK - Essential HK	Verify TC aqcceptance onboard: reception of TM (1,1)	**Not performed**	
19.	On MSTK2 window, release HIFI_report_transmission and check that the packets previously disabled appear as disabled on Enabled_Packets window (HA064289) of SCOS2000 Telemetry desktop	Verify reception of TM (1,1) and of HIFI_TM_generation_ status_report (14,4)	**Not performed**	
20.	On sSCOS2000 Telemetry desktop select ICU Housekeeping window	Check that no nominal HK are produced	**Not performed**	
21.	On sSCOS2000 Telemetry desktop select Essential Housekeeping window	Check that no essential HK are produced	**Not performed**	
22.	On MSTK2 window, release HIFI_connection test	check that no connection report is produced	**Not performed**	
23.	On MSTK2 window, release HIFI_enable_time_ver	check that no time verif. Report is produced	**Not performed**	
24.	On MSTK2 window, release HIFI_enable_transmission to re-enable the following TM packets - HIFI_Conn - Time_verif - HIFI_HK_Rev5	Verify TC aqcceptance onboard: reception of TM (1,1)	**Not performed**	
25.	On sSCOS2000 Telemetry desktop select ICU Housekeeping window	Check that nominal HK are produced	**Not performed**	
26.	On sSCOS2000 Telemetry desktop select Essential Housekeeping window	Check that no essential HK are produced	**Not performed**	
27.	On MSTK2 window, release HIFI_connection test	check that the connection report is produced	**Not performed**	
28.	On MSTK2 window, release HIFI_enable_time_ver	Check that a time verif. Report is produced	**Not performed**	
29.	Run the TOPE script “HIFI_disable_TM” to check the capability to disable all types of telemetry packets.	Verify reception of TM (1,1) and of HIFI_TM_generation_ status_report (14,4)	**Not performed**	
30.	Run the TOPE script “HIFI_Enable_TM” to check the capability to enable all types of telemetry packets.	Verify reception of TM (1,1) and of HIFI_TM_generation_ status_report (14,4)	**Not performed**	

4.3 TP3 - HK_HANDLING/Issue specific tests

Tests executed 24/02/2009. CDMS telemetry log files are stored in the OBS6.1.1 archive available at IFSI.
Not all tests listed in the standard procedure have been performed, and some extra tests have been added to check the SPR fixing. In particular:

1. SCR 2114 - The only limit check procedure that has been tested is the one involving the LO temperature monitoring: all other monitoring procedures have not been affected by the OBS 6.1.1 modifications. OBS has been modified to correctly ingest the HIFI_LOU_T_check_on TC and to implement the control of the LO temperature HK at the rate of the nominal HK acquisition. The following checks have been executed:

   A. at start up control OBS limit check against the default limits. This check has been performed by using local commands and the subsystem simulator in "commandable" mode. The check was executed to test both the upper and lower default limits and to control that the default Nbreach was taken into account correctly. The sequence of LO commands issued by OBS has been checked on the subsystem simulator log files. The test has been successful.

   B. at runtime control that the new HIFI_LOU_T_check_on is ingested and executed correctly: this check has been performed by using local commands and the subsystem simulator in "commandable" mode. The check was executed to test both the upper and lower commanded limits and to control that the commanded Nbreach was properly taken into account. The correct generation of the HIFI_LOU_T_OOL telemetry packet has been checked. The packet type, subtype have been assumed to be (5,4). The sequence of LO commands issued by OBS has been checked on the subsystem simulator log files. The test has been successful.

   C. the correct execution of an abort procedure in case of temperature out of limit detected during a measurement running has been tested by starting a HRS_tune measurement immediately before the activation of the limit check with new commanded limits. The correct implementation of the procedure has been verified by inspecting both the CDMS telemetry log files and the subsystem simulator log files. The test has been successful.

   D. at runtime control that the new HIFI_LOU_T_check_off is ingested and executed correctly: as in the previous cases this check has been performed by using local commands and the subsystem simulator in "commandable" mode. The check was executed to test that both the upper and lower commanded limits are ignored in case the limit check service has been commanded to be switched off. The test has been successful.

2. SCR 2117 on OBS 6.1.1 – All OBS events generated onboard shall have type/subtype equal to (5,1), included the limit check Out Of Limits reports. The test has been carried out by re-executing the LO temperature limit check tests and controlling that the HIFI_LOU_T_OOL telemetry packet is now generated with packet type/subtype = (5,1).

Step	Action	Expected Reaction	Observed Reaction	Notes
1.	On SCOS2000 Telemetry desktop select ICU Housekeeping window and note current HK rate		**OK - \1pkt/sec**	
2.	On MSTK2 window, release HIFI_HK_off	Verify TC aqcceptance onboard: reception of TM (1,1) Check that no nominal HK are produced	**OK**	
3.	On MSTK2 window, release HIFI_HK_on with rate =1/s and all subsystems selected	Verify reception of TM (1,1) On SCOS2000 MON1 ICU Housekeeping window check that Nominal Hk packets are produced at a rate of 1	**OK**	

		packet/sec.		
4.	Repeat step 3 selecting different active subsystems set	Verify reception of TM (1,1) On SCOS2000 MON1 ICU Housekeeping window check that Nominal Hk packets are produced at a rate of 1 packet/sec for the selected subsystem set	**OK**	
5.	On MSTK2 window, release HIFI_HK_on with rate =1/s and all subsystems selected	Verify reception of TM (1,1)	**OK**	All verifications have been carried out by activating all subsystems contemporaneously.
6.	Run the Subsyssim.tcl TOPE script to control the FCU-HK of the subsystems simulator. .	On the TCHIST window, the execution status of all 152 TC sent out shall be green. The FCU HK monitored values in MON1 AND HA 026289 window shall turn green.	**Not Performed**	
7.	On MSTK2 window, release HIFI_HK_on selecting a rate equal to the original one (step 1 of this procedure)	Verify reception of TM (1,1) On SCOS2000 MON1 ICU Housekeeping window check that Nominal Hk packets are produced at the selected rate.	**OK**	
8.	On MSTK2 window, release HIFI_non-periodic_hk_FCU_request	verify reception of TM (1,1) and of Non periodic HK report: TM (3,25), SID=0x17 to verify the content of the report: Compare the contents of the table reported in A1.3 and the values in the Non periodic hk report.	**Not Performed**	.
9.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition. Stop the subsystem simulator and save the log files of the subsystem simulator	Verify the reception of a TM(1,1)	**Not Performed**	
10.	Restart the subsystem simulator script for testcase01. (simulate increasing Values mode)		**Not Performed**	
11.	On MSTK2 window, release HIFI_housekeeping_on with rate =1/s	Verify the reception of a TM(1,1). On SCOS2000 MON1 ICU_Housekeeping window check that Nominal Hk packets are produced at a rate of 1 packet/sec.	**Not Performed**	
12.	On MSTK2 window, release HIFI_nonperiodic_hk_LCU_request with Freq = 7 Band = 7	Verify the reception of a TM(1,7) packet followed by a LCU non periodic HK report.	**Not Performed**	
13.	On MSTK2 window, release HIFI_non-periodic_hk_LCU_request with Freq = 30 Band = 7	Verify the reception of a TM(1,7) packet followed by a LCU non periodic HK report .	**Not Performed**	
14.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition. Stop the subsystem simulator and save the log files of the subsystem simulator	Verify the reception of a TM(1,1) On the simulator cmd.log file check that the LCU commands issued by OBS are equal to the reference list reported in A1.4	**Not Performed**	

15.	Restart the subsystem simulator script for testcase01.		**Not Performed**	
16.	On MSTK2 window, release HIFI_housekeeping_on with rate =1/s	Verify the reception of a TM(1,1). On SCOS2000 MON1 ICU_Housekeeping window check that Nominal Hk packets are produced at a rate of 1 packet/sec.	**Not_performed**	
17.	Run the LimitCheck.tcl TOPE script to check the autonomy functions.	Verify the acceptance of the two DHTR_CHECK on (one per each polarization) TCs and of the two single commands to set the DHTR current to 1. Verify the production of two status reports (8,6).	**Not_performed**	
18.	Wait 7 sec.	Verify that no OOL reports are issued and check the execution of the two single commands to set the DHTR current to 50001.	**Not_performed**	
19.	Wait 7 sec	Verify that no OOL reports are issued. Check the execution of the two single commands to set the DHTr current to 50002.	**Not_performed**	
20.	Wait 5 sec	Verify that two OOL reports are issued.	**Not_performed**	
21.	Wait 5 sec	Check the execution of the two single commands to set the DHTr current to 50001.	**Not_performed**	
22.	Wait 7 sec.	Verify that no OOL reports are issued. Check the execution of the two DHTR_CHECK OFF TCs and of the two single commands to set the DHTR current to 50002.	**Not_performed**	
23.	Wait 5 sec	Verify that no OOL reports are issued.	**Not_performed**	
24.	Wait 5 sec	After 5 acquisitions of nominal TM (3,25) HK packets, check that no Limit-check report is provided	**Not_performed**	
25.	Wait 5 sec	Verify the acceptance of the two WBS laser T check on (one per each polarization) TCs.	**Not_performed**	
26.	Wait 15 sec	Verify that two OOL reports are issued.	**Not_performed**	
27.	Wait 5 sec	Verify the acceptance of the two WBS laser T check OFF (one per each polarization) TCs	**Not_performed**	
28.	Wait 10 sec	Verify that no OOL reports are issued any more and Verify the acceptance of the two WBS laser T check on (one per each polarization) TCs	**Not_performed**	
29.	Wait 10 sec	Verify that no OOL reports are issued any more Verify the acceptance of the two WBS laser T check OFF (one per each polarization) TCs.	**Not_performed**	
30.	Wait 10 sec	Verify that no OOL reports are issued any more Verify the acceptance of the	**Not_performed**	

		two WBS laser T check ON (one per each polarization) TCs.		
31.	Wait 10 sec	Verify the acceptance of the two WBS laser T check OFF (one per each polarization) TCs	**Not_performed**	
32.	On the subsystem simulator set the delay time for the FCU data production to 1000.		**Not_performed**	
33.	Wait 10 sec Set the delay time for the FCU data production to 1000.	Verify that no OOL reports are issued any more Verify the acceptance of the FCUnonresponse_chk ON TC.	**Not_performed**	
34.	Wait 20 sec	Verify that the HIFI_MX_H_nonresponse and the HIFI_MX_V_nonresponse and the HIFI_Chop_nonresponse reports are issued.	**Not_performed**	
35.	Wait 20 sec	Verify the acceptance of the FCUnonresponse_chk OFF TC. Verify that no OOL reports are issued any more Verify the completion of the TOPE script.	**Not_performed**	
36.	Wait 20 sec	Verify that the Conf_FCU_power TC is issued and is immediately followed by the three non response reports	**Not_performed**	
37.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition. Stop the subsystem simulator and save the log files of the subsystem simulator	Verify the reception of a TM(1,1). On the cmd.log file verify the presence of the following commands: Mixer chain powerH OFF Mixer chain powerV OFF HWH_Laser1_OFF HWH_Laser2_OFF HWV_Laser1_OFF HWV_Laser2_OFF Mixer chain powerH OFF Mixer chain powerV OFF Chopper Board OFF	**Not_performed**	
38.	Restart the subsystem simulator script for testcase01		**Not_performed**	
39.	On MSTK2 window, release HIFI_housekeeping_on with rate =1/s	Verify the reception of a TM(1,1). On SCOS2000 MON1 ICU_Housekeeping window check that Nominal Hk packets are produced at a rate of 1 packet/sec.	**Not_performed**	
40.	Run the TP_10_1.tcl TOPE script to start a total power measurement.	Verify the reception of a TM(1,1). Verify the production of SD packets.	Not performed	
41.	On MSTK2 window, release HIFI_WH_Laser_T_chk on HIFI_WV_Laser_T_chk on	Verify the acceptance of the TCs	Not performed	
42.	Wait >5 sec	Verify that two OOL reports are issued. Note the delta time between the TC and the OOL.	Not performed	
43.	Wait 10 sec. On MSTK2 window, release HIFI_WH_Laser_T_chk off HIFI_WV_Laser_T_chk off	Verify that no OOL reports are issued. Verify the correct completion of the measurement. (NO TM(1,8) packets shall be	Not performed	

		genarted.		

4.4 TP4 - MEMORY_MANAGEMENT

These tests have not been carried out because OBS 6.1.1 doesn't contain any modification affecting the memory management service implemented onboard.

Step	Action	Expected Reaction	Observed Reaction	Notes
1.	On MSTK2 window, release HIFI_load_PRAM TC, to load a PM segment. For the TC contents refer to A2.2, command TP4.1	Verify TC acceptance and execution onboard: reception of TM (1,1) and (1,7)	N/A	
2.	On MSTK2 window, release HIFI_load_DRAM TC, to load a DM segment. For the TC contents refer to A2.2, command TP4.4	Verify TC acceptance and execution onboard: reception of TM (1,1) and (1,7)	N/A	
3.	On MSTK2 window, release HIFI_Dump_Memory TC, to dump the previously loaded PM segment For the TC contents refer to A2.2, command TP4.2	Verify TC acceptance and execution onboard: reception of TM (1,1) and (1,7) Verify the reception of one dump report (6,6) with a content equal to the one listed in A1.5	N/A	
4.	On MSTK2 window, release HIFI_Dump_Memory TC, to dump the previously loaded DM segment For the TC contents refer to A2.2, command TP4.5	Verify TC acceptance and execution onboard: reception of TM (1,1) and (1,7) Verify the reception of one dump report (6,6) with a content equal to the one listed in A1.5	N/A	
5.	On MSTK2 window, release HIFI_Check_Memory TC, to check the previously loaded PM segment For the TC contents refer to A2.2, command TP4.3	Verify TC acceptance and execution onboard: reception of TM (1,1) and (1,7) Verify the reception of a check report (6,10) with a content equal to the one listed in A1.5	N/A	
6.	On MSTK2 window, release HIFI_Check_Memory TC, to check the previously loaded DM segment For the TC contents refer to A2.2, command TP4.6	Verify TC acceptance and execution onboard: reception of TM (1,1) and (1,7) Verify the reception of a check report (6,10) with a content equal to the one listed in A1.5	N/A	
7.	On MSTK2 window, release HIFI_Dump_Memory TC, to dump a long PM segment and obtain more than one report packets. For the TC contents refer to A2.2, command TP4.8	Verify TC acceptance and execution onboard: reception of TM (1,1) and (1,7) Verify the reception of TBD dump reports (6,6) with a content equal to the one listed in A1.5 On SCOS2000 Telemetry Desktop ICU_Housekeeping window check that Spectroscopy_AID=0; HI_HK_Pool < 22;	N/A	
8.	On MSTK2 window, release HIFI_Dump_Memory TC, to dump a long DM segment and obtain more than one report packets. For the TC contents refer to A2.2, command TP4.7	Verify TC acceptance and execution onboard: reception of TM (1,1) and (1,7) Verify the reception of TBD dump reports (6,6) with a content equal to the one listed in A1.5 On SCOS2000 Telemetry Desktop ICU_Housekeeping window check that Spectroscopy_AID=0; HI_HK_Pool < 22;	N/A	

9.	On MSTK2 window, release HIFI_Dump_Memory TC, to dump a long PM segment. Immediately after, before the completion of the execution of the previous command, release a HIFI_Abort_memoryDump TC. For the TC contents refer to A2.2, command TP4.9	Verify HIFI_Dump_Memory TC acceptance: reception of TM (1,1). Verify the reception of some full length dump reports (6,6). Verify HIFI_Abort_DumpMem acceptance: reception of TM (1,1). Verify the reception of a TM (1,8) packet indicating the execution failure of the running HIFI_Dump_Memory.	N/A	
10.	Set CDMS TC source to Local Commands.		N/A	
11.	Send PM_wrong_MID to test the OBS reaction against a wrong Memory ID.	Verify reception of TM (1,2) Invalid memory ID error code 0x1002	N/A	
12.	Send PM_wrong_sadd to test the OBS reaction against a wrong start address.	Verify reception of two TM (1,2) Invalid Start Address error code 0x1003.	N/A	
13.	Send PM_wrong_size to test the OBS reaction against the attempt to write out of memory.	Verify reception of TM (1,2) Invalid memlength error code 0x1000	N/A	
14.	Send PM_wrong_length to test the OBS reaction against a wrong number of data words in the length field (in the Application Data)	Verify reception of TM (1,2) Invalid memlength error code 0x1000	N/A	
15.	Send PM_wrong_crc to test the OBS reaction against a wrong CRC checksum for the uplinked memory patch (not the CRC of the whole TC).	Verify reception of TM (1,2) Invalid crc error code 0x1003	N/A	
16.	Send DM_wrong_MID to test the OBS reaction against a wrong Memory ID.	Verify reception of TM (1,2) Invalid memory ID error code 0x1002	N/A	
17.	Send DM_wrong_sadd to test the OBS reaction against a wrong start address.	Verify reception of two TM (1,2) Invalid Start Address error code 0x1003.	N/A	
18.	Send DM_wrong_size to test the OBS reaction against the attempt to write out of memory.	Verify reception of TM (1,2) Invalid memlength error code 0x1000	N/A	
19.	Send DM_wrong_length to test the OBS reaction against a wrong number of data words in the length field (in the Application Data)	Verify reception of TM (1,2) Invalid memlength error code 0x1000	N/A	
20.	Send DM_wrong_crc to test the OBS reaction against a wrong CRC checksum for the uplinked memory patch (not the CRC of the whole TC).	Verify reception of TM (1,2) Invalid crc error code 0x1003	N/A	
21.	Send Check_PM TC	Verify reception of TM (1,1), (1,7) Verify the reception of a check report (6,10) with a content variable with the OBS version.	N/A	
22.	Verify the procedure to patch OBS.	TBW	N/A	
23.	Set CDMS TC source to Router		N/A	

4.5 TP5 - CONFIGURE_SUBSYSTEMS

These tests have not been carried out because both OBS 6.0 and OBS 6.1 don't contain any modification affecting the configure subsystems service implemented onboard.

Step	Action	Expected Reaction	Observed Reaction	Notes
1.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition. Stop the subsystem simulator and save the log files of the subsystem simulator	Verify the reception of a TM(1,1)	N/A	
2.	Restart the subsystem simulator script for testcase01. To store cmd.log: tail –f cmd.log | annotateCommands.csh nopreps=1 notrans=1		N/A	
3.	On MSTK2 window, release HIFI_housekeeping_on with rate =1/s	Verify the reception of a TM(1,1). On SCOS2000 MON1 ICU_Housekeeping window check that Nominal Hk packets are produced at a rate of 1 packet/sec.	N/A	
4.	Run configureSubsystems.tcl TOPE script.	Check the status of the commands in TCHIST. The columns A and C should be ticked with S and have the green color	N/A	
5.	Wait 5 sec		N/A	
6.	Run HIFI_Single_cmd_simulator.tcl TOPE script.	Check the status of the commands in TCHIST. The columns A and C should be ticked with S and have the green color	N/A	
7.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition. Stop the subsystem simulator and save the log files of the subsystem simulator Store the output of the annotate-script and compare this with the reference data reported in A1.6.	the reference data reported in A1.6 contain also the reference data for HIFI_single command script	N/A	.
8.	Restart the subsystem simulator script for testcase01		N/A	
9.	On MSTK2 window, release HIFI_housekeeping_on with rate =1/s	Verify the reception of a TM(1,1). On SCOS2000 MON1 ICU_Housekeeping window check that Nominal Hk packets are produced at a rate of 1 packet/sec.	N/A	
10.	Set CDMS TC source to Local Commands.		N/A	This step is needed to send wrong TCs.
11.	Send HIFI_Configure_FCU_err2 to test for a wrong (minor) number of parameters	Verify the reception of a TM(1,2) packet with "NOK_CMDSEQ_ILLEGAL _APPLICATION_DATA" err ID. Verify the reception of a TM(5,4) EV packet with "NOK_CMDSEQ_LENGTH _SECOND_CK" EV ID, with two parameters, the actual length and the expected length..	N/A	

12.	Send HIFI_Configure_FCU_err3 to test for a wrong (in excess) number of parameters	Same as step 11	N/A	
13.	Send HIFI_Configure_FCU_power_err2 to test for a wrong (minor) number of parameters	Same as step 11	N/A	
14.	Send HIFI_Configure_FCU_power_err3 to test for a wrong (in excess) number of parameters	Same as step 11	N/A	
15.	Send HIFI_Config_WBS_H_err2.txt to test for a wrong (minor) number of parameters	Same as step 11	N/A	
16.	Send HIFI_Config_WBS_H_err3.txt to test for a wrong (in excess) number of parameters	Same as step 11	N/A	
17.	Send HIFI_Config_HRS_H_att_lo_err2.txt to test for a wrong (minor) number of parameters	Same as step 11	N/A	
18.	Send HIFI_Config_HRS_H_att_lo_err3.txt to test for a wrong (in excess) number of parameters	Same as step 11	N/A	
19.	Send HIFI_Configure_LCU1a_err2.txt to test for a wrong (minor) number of parameters	Same as step 11	N/A	
20.	Send HIFI_Configure_LCU1a_err3.txt to test for a wrong (in excess) number of parameters	Same as step 11	N/A	
21.	Set CDMS TC source to Router		N/A	

4.6 TP6 – SPECTROSCOPY Measurements

Assuming that the minor patch implemented in OBS 6.1.1 has no impact on the functionalities checked with TP6, noTP6 tests have been performed on OBS 6.1.1.

Step	Action	Expected Reaction	Observed Reaction	Notes
1.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition.	Verify the reception of a TM(1,1)	Not Performed	
2.	In the subsystem simulator kill the presently running script.	The running processes windows shall disappear.	Not Performed	
3.	In the subsystem simulator run the script for ifsi_ testcase01. To store cmd.log: tail –f cmd.log | annotateCommands.csh nopreps=1 notrans=1	All 6 processes windows shall appear.	Not Performed	
4.	Restart the HIFI_packet_logger to have a separate directory to store the Total Power test data.		Not Performed	
5.	On MSTK2 window, release HIFI_housekeeping_on with rate =1/s	Verify the reception of a TM(1,1). On SCOS2000 MON1 ICU_Housekeeping window check that Nominal Hk packets are produced at a rate of 1 packet/sec. Verify the packet acquisition in the HIFI-packet_logger terminal.	Not Performed	
6.	On TOPE run Total_Power tcl script. It includes the execution of 12 TP measurements with variable input parameters. See Table Table A2.3 .1/2 to know the configuration of each measurement. The script is reported in A3.5	Verify the reception of a Configure spectroscopy TC and of a HIF_Configure spectroscopy_report (8,6)	Not Performed	TOTAL POWER TEST The steps from 6 to 32 verify the items 2, 4 and 5 of section 5.6 of SVVP.
7.	On Telemetry Desktop select HIFI_configure_spectroscopy_e cho log window	Check that the parameters are equal to those reported in row 1 of Table A2.3 .1.	Not Performed	Measure 1:
8.	Wait 3 sec	Verify the reception of a start Total power TC. Check the status of the commands in TCHIST. Verify the science data packet acquisition in the HIFI-packet_logger terminal On SCOS2000 Telemetry Desktop ICU_Housekeeping window check that Spectroscopy_AID=1; HI_SD_Pool < 30; VM_RUNNING=1;	Not Performed	
9.	Wait 15 sec (end of meas 1)	Verify the reception of the successful execution completion report (1,7). Check the status of the command in TCHIST (all green entries) . On SCOS2000 Telemetry Desktop ICU_Housekeeping window check that Spectroscopy_AID=0; HI_SD_Pool =8;	Not Performed	The analysis of the science data can be done at the end of the execution of the whole TOPE procedure.

10.	Wait 3 sec	As in Step 8. Meas 2 Configuration Parameters can be monitored on Telemetry Desktop HIFI_configure_spectroscopy _echo log window	Not Performed	Measure 2
11.	Wait 15 sec (end of meas 2)	As in Step 8.	Not Performed	As in Step 8.
12.	Wait 3 sec	As in Step 10.	Not Performed	Measure 3
13.	Wait 15 sec (end of meas 3)	As in Step 8.	Not Performed	As in Step 8.
14.	Wait 3 sec	As in Step 10.	Not Performed	Measure 4
15.	Wait 15 sec (end of meas 4)	As in Step 8.	Not Performed	As in Step 8.
16.	Wait 3 sec	As in Step 10.	Not Performed	Measure 5
17.	Wait 15 sec (end of meas 5)	As in Step 8.	Not Performed	As in Step 8.
18.	Wait 3 sec	As in Step 10.	Not Performed	Measure 6
19.	Wait 15 sec (end of meas 6)	As in Step 8.	Not Performed	As in Step 8.
20.	Wait 3 sec	As in Step 10.	Not Performed	Measure 7
21.	Wait 25 sec (end of meas 7)	As in Step 8.	Not Performed	As in Step 8.
22.	Wait 3 sec	As in Step 10.	Not Performed	Measure 8
23.	Wait 25 sec (end of meas 8)	As in Step 8.	Not Performed	As in Step 8.
24.	Wait 3 sec	As in Step 10.	Not Performed	Measure 9
25.	Wait 25 sec (end of meas 9)	As in Step 8. Verify that No science data are provided at all (0 range selected for all spectrometers) On ICU HK verify that HI_SD_Pool =8	Not Performed	As in Step 8.
26.	Wait 3 sec	As in Step 10.	Not Performed	
27.	Wait 25 sec (end of meas 10)	As in Step 8. Verify that only a subsample of the total number of SD packets is received (See A1.7, Table 1)	Not Performed	As in Step 8.
28.	Wait 3 sec	As in Step 10.	Not Performed	
29.	Wait 25 sec (end of meas 11)	As in Step 8.	Not Performed	As in Step 8.
30.	Wait 3 sec	As in Step 10.	Not Performed	
31.	Wait 25 sec (end of meas 12)	As in Step 8.	Not Performed	As in Step 8.
32.	Stop HIFI_packet_logger and analyse results.	Verify the compatibility of the results with the contents of A1.7, Tables 1 and 2. To check the spectra, use the script Diffe.bat. Verify that the latched time reported in the start data frames is compatible with the commanded integration times.	Not Performed	
33.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition.	Verify the reception of a TM(1,1)	Not Performed	
34.	In the subsystem simulator kill the presently running script.	The running processes windows shall disappear.	Not Performed	
35.	In the subsystem simulator run the script for ifsi_ testcase02. To store cmd.log: tail –f cmd.log |	All 6 processes windows shall appear.	Not Performed	

	annotateCommands.csh nopreps=1 notrans=1			
36.	Restart the HIFI_packet_logger to have a separate directory to store the Fast Chop test data.		Not Performed	
37.	On TOPE run the HIFI_Fast_Chop.tcl script. It includes the execution of 2 FC measurements with variable input parameters. See Table Table A2.3 .3/4 to know the configuration of each measurement. The script is reported in A3.5	Verify the reception of a Configure spectroscopy TC and of a HIF_Configure spectroscopy_report (8,6)	Not Performed	
38.	On Telemetry Desktop select HIFI_configure_spectroscopy_echo log window	Check that the parameters are equal to those reported in row 1 of Table A2.3 .1.	Not Performed	
39.	Wait 3 sec	Verify the reception of a start Fast Chop TC. Check the status of the commands in TCHIST. Verify the science data packet acquisition in the HIFI-packet_logger terminal On SCOS2000 Telemetry Desktop ICU_Housekeeping window check that Spectroscopy_AID=2; HI_SD_Pool < 30; VM_RUNNING=1;	Not Performed	
40.	Wait 40 sec (end of meas 1)	Verify the reception of the successful execution completion report (1,7). Check the status of the command in TCHIST (all green entries) . On SCOS2000 Telemetry Desktop ICU_Housekeeping window check that Spectroscopy_AID=0; HI_SD_Pool =8;	Not Performed	
41.	Wait 3 sec	As in step 39.	Not Performed	
42.	Wait 40 sec (end of meas 2)	As in step 40	Not Performed	
43.	Stop HIFI_packet_logger and analyse results.	Verify the compatibility of the results with the contents of A1.7, Tables 3 and 4. To check the spectra, use the script Diffe.bat. Verify that the latched time reported in the start data frames is compatible with the commanded integration times.	Not Performed	
44.	SLOW CHOP TEST. TBW		Not performed	
45.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition.	Verify the reception of a TM(1,1)	Not performed	
46.	In the subsystem simulator kill the presently running script.	The running processes windows shall disappear.		
47.	In the subsystem simulator run the script for testcase01. To store cmd.log: tail –f cmd.log | annotateCommands.csh nopreps=1 notrans=1	All 6 processes windows shall appear.		

48.	On MSTK2 window, release a HIFI_Configure_spectroscopy for a long duration Total Power and issue the Start_Total Power Command.	Verify the Acceptance of both TCs and the reception of the Configure echo packet (8,6). On SCOS2000 Telemetry Desktop ICU_Housekeeping window check that Spectroscopy_AID=1; HI_SD_Pool < 30; VM_RUNNING=1;	Not performed	
49.	On MSTK2 window, release a HIFI_Abort_spectroscopy Command.	Verify the Acceptance of the TC and the reception of the TC execution failure TM report (1,8) referred to the running measurement. On SCOS2000 Telemetry Desktop ICU_Housekeeping window check that while aborting the Spectroscopy_AID=1000; After the abort completion: Spectroscopy_AID=0; HI_SD_Pool < 30; VM_RUNNING=1;	Not performed	
50.	Restart the HIFI_packet_logger to have a separate directory to store the Full Performance Total Power test data. Select the FIFO tester and the ECHO tester functionalities.		Not performed	
51.	On TOPE run the HIFI_Full _performance.tcl script. See section 3.5 of AD13 (SRON Test Procedure) for a description of the test The script is reported in A3.5	Verify the reception of a Configure spectroscopy TC and of a HIF_Configure spectroscopy_report (8,6)	Not performed	
52.	Wait 3 sec	Verify the reception of the Start Total Power TC. Check the status of the commands in TCHIST. Verify the science data packet acquisition in the HIFI-packet_logger terminal On SCOS2000 Telemetry Desktop ICU_Housekeeping window check that Spectroscopy_AID=1; HI_SD_Pool < 30; VM_RUNNING=1;	Not performed	
53.	Wait 1h.	Verify the reception of the successful execution completion report (1,7). Check the status of the command in TCHIST (all green entries) . On SCOS2000 Telemetry Desktop ICU_Housekeeping window check that Spectroscopy_AID=0; HI_SD_Pool =8;	Not performed	
54.	Stop HIFI_packet_logger and analyse results.	See section 3.5 of of AD13 (SRON Test Procedure) for a description of the checks to be done.	Not performed	
55.	In the subsystem simulator analyse the simulator log files:	Verify that the maximum jitter during the measurement	Not performed	

	calculate the time-interval of the successive WBS-starts, and the corresponding maximum jitter. Make use of the script jitter.awk	is less that 3usec. Verify that the acquisition of the nominal HK from FCU and LCU doesn't afftect the measurement timing.		
56.	On MSTK2 window, release a HIFI_Notify_PDU_status to simulate the switch off WBS_H and HRS_V.		Not performed	
57.	Repeat steps 1-32.	Verify that the measurements results are compliant with the expected results for the TP procedure. Verify that only WBSV and HRSH data are present. During this procedure, only half of the expected packet sequences reported in Table A1.7.1should be expected.	Not performed	
58.	On the CDMS simulator switch to Local Commands. Refer to A2.4 for a description of the Commands to be used.		Not performed	
59.	Issue command **conf_spect_new_err_1.**	Verify reception of TM(1,2) (acceptance failure) with error code 0x5 (invalid application data) followed by a type,subtype 5,4 packet (runtime error) with error code 0x10 (error on packet len) and two parameters: the first parameter value shall be 004C (the actual, wrong lenght) and the second parameter value shall be 004E (the expected, correct length for the configure spectroscopy command).	Not performed	
60.	Issue command **conf_spect_new_err_2.**	Verify reception of TM(1,2) (acceptance failure) with error code 0x0626 (wbs illegal accumulation time) and no parameters.	Not performed	

4.7 TP7 - PARAMETERS_SCAN and other extra tests

The purpose of this test is to verify the capability of the OBS to execute the FCU/LCU Scan functions and the LO scan and tune commands.
In case of OBS 6.1.1 no TP7 tests have been carried out because the implemented code modifications have no impact on the parameter scan service.

In this test scenario the following functionalities will be tested :

1. Verification of the correct execution of a FCU parameter scan (HIFI_FCU_PARAMETER_SCAN TC) with the new type of input parameters as per SCR 1689:
    a. Verification of the correct handling of a FCU parameter scan in case of commanded negative steps;
    b. Verification of the correct handling of a FCU parameter scan in case of commanded positive steps;
    c. Verification of the correct handling of a FCU parameter scan in case of a commanded null number of N magnet;
    d. Verification of the correct handling of a FCU parameter scan in case of a commanded null number of N voltage;

Not performed.

2. Verification of the correct execution of a LCU parameter scan (HIFI_LCU_IV_curve TC) in the nominal case;
    a. Verification of the correct handling of a LCU parameter scan in case of a wrong commanded total number of parameters;

Not performed.

3. Verification of the correct execution of a HIFI_Sweep_Diplexer_without_Ifpower with the new type of input parameters as per SCR 1727:
    a. Verification of the correct handling of a HIFI_Sweep_Diplexer_without_Ifpower TC in case of commanded negative steps;
    b. Verification of the correct handling of a HIFI_Sweep_Diplexer_without_Ifpower TC in case of commanded positive steps;
    c. Verification of the correct handling of a HIFI_Sweep_Diplexer_without_Ifpower TC in case of a maximum number of steps greater than the maximum allowed;

Not performed.

4. Verification of the correct execution of a HIFI_Sweep_Diplexer_with_Ifpower with the new type of input parameters as per SCR 1727:
    a. Verification of the correct handling of a HIFI_Sweep_Diplexer_with_Ifpower TC in case of commanded negative steps;
    b. Verification of the correct handling of a HIFI_Sweep_Diplexer_with_Ifpower TC in case of commanded positive steps;
    c. Verification of the correct handling of a HIFI_Sweep_Diplexer_with_Ifpower TC in case of a maximum number of steps greater than the maximum allowed;

Not performed.

5. Verification of the correct execution of a HIFI_Load_Vector_Scan TC to configure a LO vector scan in the nominal case
    a. Verification of the correct handling of a HIFI_Load_Vector_Scan TC to configure a LO vector scan in case of a wrong number of steps;
6. Verification of the correct execution of a LO vector scan (HIFI_vector_scan TC) in the nominal case
7. Verification of the correct execution of a LO tuning (HIFI_Tune_LO_Using_MXCH TC)

Not performed.

8. Verification of the correct execution of a Engineering Scan with the new specification in SCR 1688:
    a. verification that the engineering scan is performed correctly with the input parameters in the nominal ranges;
    b. verification that the engineering scan is performed in according to new specs. when a dummy command 0x0fffffff is used as input parameter.

Not performed.

4.8 TP8 - TUNE

The purpose of this test case is to verify the capability of the OBS to execute the tuning procedures described in AD11.
In case of OBS 6.1.1 the implemented modifications have no impact on the TP7 tuning functions and therefore have not been tested.

Step	Action	Expected Reaction	Observed Reaction	Notes
1.	On MSTK2 window, release HIFI_notify_PDU_status to switch on all spectrometers	Verify the reception of a TM(1,1)	N/A	
2.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition.	Verify the reception of a TM(1,1)	N/A	
3.	In the subsystem simulator kill the presently running script.	The running processes windows shall disappear.	N/A	
4.	In the subsystem simulator run the script for ifsi_ tunetest01. To store cmd.log: tail –f cmd.log | annotateCommands.csh nopreps=1 notrans=1	All 6 processes windows shall appear.	N/A	
5.	Restart the HIFI_packet_logger to have a separate directory to store the HRS tune test data.		N/A	
6.	On MSTK2 window, release HIFI_HRS_Tune .	On the CDMS TM log window verify reception of TM (1,1). In Telemetry Desktop ICU HK window the Spectroscopy _AID shall assume the two values 18 (tune first step) and 33 (tunesecond step).	N/A	HRS_tune test. Nominal case
7.	On the MON1 window of SCOS2000, select HRS H tuning HRS V tuning windows. The Tuning parameters acquired by SCOS2000 will be displayed on the window.	Two tune reports, one per each polarization shall be provided. The full set of parameters listed on the window shall be updated twice.	N/A	
8.	On the HIFI_Packet logger	Verify the reception of 2 HRS full packetisations per polarisation. Verify the reception of 4 tune reports. The Science Data and tune report content shall be compatible with the test results reported in A1.8 for the Nominal case.	N/A	
9.	On MSTK2 window, release HIFI_WBS_Tune .	On the CDMS TM log window verify reception of TM (1,1).	N/A	WBS_tune test. Nominal case
10.	On the MON1 window of SCOS2000, select WBS H tuning and WBSV tuning windows. The Tuning parameters acquired by SCOS2000 will be displayed on	Two tune reports, one per each polarization shall be provided. The full set of parameters listed on the window shall be updated twice.	N/A	90,50,70,30,10,15

	the window.			
11.	On the HIFI_Packet logger	Verify the reception of 3 WBS full packetisations per polarisation. Verify the reception of 2 tune reports per packetisation The Science Data and tune report content shall be compatible with the test results reported in **Error! Reference source not found.** for the Nominal case.	N/A	
12.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition.	Verify the reception of a TM(1,1)	N/A	
13.	In the subsystem simulator kill the presently running script.	The running processes windows shall disappear.	N/A	
14.	In the subsystem simulator run the script for ifsi_ tunetest02.	All 6 processes windows shall appear.	N/A	HRS/WBS tune: low signal case.
15.	Repeat steps 6-11		N/A	Reference data are those for Low Sihnal Case
16.	On MSTK2 window, release HIFI_notify_PDU_status to switch off both HRS.	Verify the reception of a TM(1,1) In Telemetry Desktop ICU HK window the HI_HRSH_HK_S HI_HRSV_HK_S parameters shall become INVALID.	N/A	
17.	Repeat step 6	Verify the reception of a TM(1,8) (execution failure) with the Err code: EXF_HS_LIB_HRS_SUB_OFF	N/A	
18.	Repeat steps 16,17 switching off WBS spectrometers.	Verify the reception of a TM(1,8) (execution failure) with the Err code: EXF_HS_LIB_HRS_SUB_OFF	N/A	
19.	Repeat step 1 and send HIFI_Housekeeping on to restart nominal HK acquisition at a rate of 1/sec.	Verify acceptance of both TCs. Verify that all spectrometers are on and that their status is VALID.	N/A	
20.	Repeat step 6 and 9	Verify the acquisition of all expected science data and reports. On the Telemetry desktop ICU Hk window, check that: Spectroscopy_AID=18, 33 (during the HRS tuning test) And Spectroscopy_AID=19,34,35 (during theWBS tuning test) HI_SD_Pool<30;	N/A	To check the spectrometers tuning in case of running HK. No data check is foreseen. Only a check that both activities can run concurrently onboard with no conflicts. Once optimised the Tune environment for the subsystem simulator, the steps 6/9 and 20 can be reduced to one step only.
21.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition.	Verify the reception of a TM(1,1)	N/A	
22.	In the subsystem simulator kill the presently running script.	The running processes windows shall disappear.	N/A	
23.	In the subsystem simulator run the script for ifsi_tunetest04.	All 6 processes windows shall appear.	N/A	Mixer Magnet Tune
24.	Send HIFI_Tune_mxmgc_useHRS TC with the following parameters: HIF_step_time= 10 (1sec) HIF_Nmagnet = 5 HIF_ch1_mx_mg0_C =default	Verify TC acceptance. Verify the reception of 1mixer magnet report per each polarization. (mixMagnetCurrent_useHRS report).	N/A	Step description to be completed

	HIF_cv1_mx_mg0_C =def HIF_mx_mg_step_C = 10	Check the report content against the expected result reported in A1.10		
25.	Send HIFI_Tune_mxmgc_useWBS TC with the following parameters: HIF_step_time= 10 (1sec) HIF_Nmagnet = 5 HIF_ch1_mx_mg0_C =default HIF_cv1_mx_mg0_C =def HIF_mx_mg_step_C = 10	Verify TC acceptance. Verify the reception of 1mixer magnet report per each polarization. (mixMagnetCurrent_useWBS report). Check the report content against the expected result reported in A1.10	N/A	Step description to be completed
26.	Run the annotate_command script on the subsystem simulator.	Analyse the content of the cmd an Hk log files of the subsystem simulator and compare them with the expected cmd/Hk request list (see A1.10)	N/A	Step description to be completed
27.	On MSTK2 window, release HIFI_housekeeping_on command to restart Hk acquisition. At a rate of 1pkt/sec.	Verify the command acceptance and execution.	N/A	
28.	Repeat steps 25 and 26 .	On the Telemetry desktop ICU Hk window, check that: Spectroscopy_AID=21 (during the test) HI_SD_Pool<30;	N/A	To check the spectrometers tuning in case of running HK. No data check is foreseen. Only a check that both activities can run concurrently onboard with no conflicts.
29.	On MSTK2 window, release HIFI_notify_PDU_status to switch off the spectrometers.	Verify the command acceptance and execution.	N/A	
30.	Repeat steps 25 and 26 .	For both steps: Verify the reception of a TM(1,8) (execution failure) with the Err code: EXF_HS_LIB_HRS_SUB_OFF	N/A	
31.	On MSTK2 window, release HIFI_housekeeping_off to stop nominal HK acquisition.		N/A	
32.	Change the HIFI packet logger results directory.		N/A	
33.	Kill the subsystem simulator running script and run the ifsi_testcase01		N/A	
34.	On MSTK2 window, release HIFI_notify_PDU_status to switch on the spectrometers.	Verify the command acceptance and execution.	N/A	
35.	On MSTK2 window, release HIFI_WBS_COMB to execute a COMB spectrum.	Verify the TC acceptance.	N/A	
36.	On Telemetry desktop select ICU HK window	Verify that during the measurement Spectroscopy_AID =38 (first step) Spectroscopy_AID=39 (second step).	N/A	
37.	Wait 5 sec.	Verify that one full packetisation for both WBS spectrometers has been received. Use the execpted results of Total power measure 1 as reference values(A1.7.1/2).	N/A	
38.	On subsystem simulator use annotate commands on the cmd and hk log files.	Verify that the correct sequence of commands has been sent out by OBS with the correct timing. Use the table in A1.11.1 for the reference data.	N/A	
39.	On MSTK2 window, release HIFI_WBS_ZERO to execute a COMB spectrum.	Verify the TC acceptance.	N/A	

40.	On Telemetry desktop select ICU HK window	Verify that during the measurement Spectroscopy_AID =40.	N/A	
41.	Wait 5sec.	Verify that one full packetisation for both WBS spectrometers has been received. Use the execpted results of Total power measure 1 as reference values(A1.7.1/2).	N/A	
42.	On subsystem simulator use annotate commands on the cmd and hk log files.	Verify that the correct sequence of commands has been sent out by OBS with the correct timing. Use the table in A1.11.2 for the reference data.	N/A	

4.9 TP9 - Peak Up

These tests have not been carried out because OBS 6.1.1 doesn't contain any modification affecting the peak up service implemented onboard.

A1. Appendix: Expected results

A1.1 ICU Housekeeping

The detailed structure of the HK packet is described in AD10, section 3.1. and reported below:

Start Byte	Start bit	Length	Monitor Parameter Description	Acceptance criterium
26	0	24	HI_SW_Version Version number of the OBS	= to the version of the OBS under test
29	0	8	HI_SW_Revision Revision number of the OBS	= to the revision of the OBS under test
30	0	32	HI_IDLE Number of loops in a second performed by the res_chk task (the lowest priority task).	
34	0	32	HI_CPU_Load_Min Minimum delay (in one sec) in msec from one loop and the next one in res_chk task.	
38	0	32	HI_CPU_Load_AV Total delay (sum) from subsequent loops n res_chk task. Must be divided by the # of loops to get the average delay.	
42	0	32	HI_CPU_Load_Max Maximum delay (in one sec) in msec from one loop and the next one in res_chk task.	
46	0	32	HI_EV_POOL Max # of blocks taken in Event Pool	<28 (= 0 at startup)
50	0	32	HI_HK_POOL Max # of blocks taken in HK Pool	<22 (= 4 at startup)
54	0	32	HI_SD_POOL Max # of blocks taken in Science Pool	<30 (= 8 at startup)
58	0	32	HI_TC_POOL Max # of blocks taken in TC Pool	<6 (= 0 at startup)
62	0	32	HI_LS_QUEUE_MAX Max depth reached in LS Queue (Virtuoso FIFO)	<512 (0x200)
66	0	32	HI_HK_QUEUE_MAX Max depth reached in HK Queue (Virtuoso FIFO)	<24 (0x18)
70	0	32	HI_SD_QUEUE_MAX Max depth reached in Science Queue (Virtuoso FIFO)	<728 (0x2d8)
74	0	32	HI_EV_QUEUE_MAX Max depth reached in Event Queue (Virtuoso FIFO)	<24 (0x18)
78	0	32	HI_TC_QUEUE_MAX Max depth reached in TC Queue (Virtuoso FIFO)	<4 (0x04)
82	0	32	HI_ER_QUEUE_MAX Max depth reached in Error Queue (Virtuoso FIFO)	<64 (0x40)
86	0	32	HI_VM_RUNNING_S	True if VM is running. False if it is stopped . (= 0 at startup).
90	0	32	HI_2P5_V 2.5 Volt actual value	N/A to AVM1 For FM see A1.2
94	0	32	HI_5P_V 5 Volt actual value	N/A to AVM1 For FM see A1.2
98	0	32	HI_15P_V 15 Volt actual value	N/A to AVM1 For FM see A1.2
102	0	32	HI_15M_V minus 15 Volt actual value	N/A to AVM1 For FM see A1.2
106	0	32	HI_CPU_T CPU Temperature	N/A to AVM1 For FM see A1.2
109	0	32	HI_SUBSYSTEM_S Current Subsystem Status	Equal to the commanded Subsystem Status Word = 0 at startup
109	2	1	HI_FCU_S FCU- subsystem status	
109	3	1	HI_LCU_S LCU- subsystem status	
109	4	1	HI_WBSV_S WBS-H status	
109	5	1	HI_WBSH_S WBS-V status	
109	6	1	HI_HRSV_S HRS-H status	
110	7	1	HI_HRSH_S HRS-V status	
114	0	32	HI_HP_QUEUE_MAX Max depth reached in LS cmd queue (Virtuoso FIFO)	<512 (0x200)
118	0	32	HI_Spectr_HK_valid Spectrometer Housekeeping validity flags	1 = spectr. data in HK 0 = spectr. data NOT in HK (= 0 at startup)
121	4	1	HI_WBSV_HK_S HK validity	
121	5	1	HI_WBSH_HK_S HK validity	
121	6	1	HI_HRSV_HK_S HK validity	
121	7	1	HI_HRSH_HK_S HK validity	
122	0	32	AID_spectroscopy	AID of the presently running activity

				(= 0 at startup)
126	0	32	HICU_HK_25 Spare	
130	0	32	HICU_HK_26 Spare	
134	0	32	HICU_HK_27 Spare	
138	0	32	HICU_HK_28 Spare	
142	0	32	HICU_HK_29 Spare	
146	0	32	HICU_HK_30 Spare	
150	0	32	HICU_HK_31 Spare	

A1.2 ICU Hardware parameters details

VOL_25P 2.5 V reference voltage. The allowed variability is ± 30% (in digital units the allowed range is [1434,2663]).

VOL_5P the output of the 5 V analogical channel. The allowed variability is ± 30% (in digital units the allowed range is [3236,3577]).

VOL_15P the output of the +15 V analogical channel. The allowed variability is ± 30% (in digital units the allowed range is [3236,3577]).

VOL_15N the output of the -15 V analogical channel. The allowed variability is ± 30% (in digital units the allowed range is [3236,3577]).

A1.3 FCU Non Periodic HK addresses

HK address	Single command word	Simulated HK value
0x8C13	CC130001	1
0x8CA8	CCA80002	2
0x8CA9	CCA90003	3
0x8CAB	CCAB0004	4
0x8CAC	CCAC0005	5
0x8CAE	CCAE0006	6
0x8CAF	CCAF0007	7
0x8CB1	CCB10008	8
0x8CB2	CCB20009	9
0x8CB4	CCB4000A	10
0x8CB5	CCB5000B	11
0x8D13	CD13000C	12
0x8DA8	CDA8000D	13
0x8DA9	CDA9000E	14
0x8DAB	CDAB000F	15
0x8DAC	CDAC0010	16
0x8DAE	CDAE0011	17
0x8DAF	CDAF0012	18
0x8DB1	CDB10013	19
0x8DB2	CDB20014	20
0x8DB4	CDB40015	21
0x8DB5	CDB50016	22
0x8F11	CF110017	23
0x8F13	CF130018	24
0x8F14	CF140019	25
0x8F15	CF15001A	26
0x8F16	CF16001B	27
0x8F17	CF17001C	28
0x8F18	CF18001D	29
0x8F19	CF19001E	30
0x8F26	CF26001F	31
0x8C20	CC200020	32
0x8C22	CC220021	33
0x8D20	CD200022	34

0x8D22	CD220023	35
0x8F10	CF100024	36
0x8C10	CC100025	37
0x8D10	CD100026	38
0x8C24	CC240027	39
0x8D24	CD240028	40

A1.4 LCU Non Periodic HK requests

Freq=7, band =7
Expected commands:
0xf10f2087
0xf10f0007
0xf10f2007
0xf10f2097
0xf10f4097
0xf10f0017
0xf10f4017
0xf10f2017
0xf10f6717
0xf10f6797
0xf10f20a7
0xf10f40a7
0xf10f0027
0xf10f4027
0xf10f2027
0xf10f6727
0xf10f67a7
0xf10f20b7
0xf10f40b7
0xf10f0037
0xf10f4037
0xf10f2037
0xf10f6737
0xf10f67b7
0xf10f20c7
0xf10f40c7
0xf10f0047
0xf10f4047
0xf10f2047
0xf10f67c7
0xf10f6747
0xf10f20d7
0xf10f40d7
0xf10f0057
0xf10f4057
0xf10f2057
0xf10f67d7
0xf10f6757
0xf10f20e7
0xf10f40e7
0xf10f0067
0xf10f4067
0xf10f2067
0xf10f67e7
0xf10f6767
0xf10f20f7
0xf10f47f7

0xf10f0077
0xf10f4077
0xf10f2077
0xf10f67f7
0xf10f6777
0xf10f8787
0xf10f8707
0xf10fa707

Freq=30, band =7
Expected commands:
0xf10f2087
0xf10f0007
0xf10f2007
0xf10f2097
0xf10f4097
0xf10f0017
0xf10f4017
0xf10f2017
0xf10f7e17
0xf10f7e97
0xf10f20a7
0xf10f40a7
0xf10f0027
0xf10f4027
0xf10f2027
0xf10f7e27
0xf10f7ea7
0xf10f20b7
0xf10f40b7
0xf10f0037
0xf10f4037
0xf10f2037
0xf10f7e37
0xf10f7eb7
0xf10f20c7
0xf10f40c7
0xf10f0047
0xf10f4047
0xf10f2047
0xf10f7ec7
0xf10f7e47
0xf10f20d7
0xf10f40d7
0xf10f0057
0xf10f4057
0xf10f2057
0xf10f7ed7
0xf10f7e57
0xf10f20e7
0xf10f40e7
0xf10f0067
0xf10f4067
0xf10f2067
0xf10f7ee7

0xf10f7e67
0xf10f20f7
0xf10f5ef7
0xf10f0077
0xf10f4077
0xf10f2077
0xf10f7ef7
0xf10f7e77
0xf10f9e87
0xf10f9e07
0xf10fbe07

A1.5 Memory Management

Proc. step	Ref. Tc	Mnemonic TC	Expected result
Step 1	TP4.1	**mem_load_PM**.	
Step 2	TP4.4	**mem_load_DM**.	
Step 3	TP4.2	**mem_dump_PM**.	0006 0000 0004 AAAA BBBB CCCC AAAA BBBB CCCC AAAA BBBB CCCC AAAA BBBB CCCC B3BA
Step 4	TP4.5	**mem_dump_DM**.	0102 2000 0004 AAAA BBBB AAAA BBBB AAAA BBBB AAAA BBBB 0446
Step 5	TP4.3	**mem_check_PM**.	0006 0000 0004 B3BA
Step 6	TP4.6	**mem_check_DM**.	0102 2000 0004 0446
Step 7	TP4.8	**mem_dump_long_PM**.	**TBW**
Step 8	TP4.7	**mem_dump_long_DM**.	**TBW**
Step 9	TP4.9	**mem_dump_abort**.	

A1.6 TP5 Configure Subsystems

Configure_FCU:

0xcf010001	HF_CPR_MXBAND
0xcc1300e3	HF_CH1_DPFPP1
0xcca80000	HF_CH2_FIF1_Drain_V
0xcca90000	HF_CH2_FIF1_Drain_C
0xccab0000	HF_CH2_FIF2_Drain_V
0xccac0000	HF_CH2_FIF2_Drain_C
0xccae0000	HF_CH2_SIF1_Drain_V
0xccaf0000	HF_CH2_SIF1_Drain_C
0xccb10000	HF_CH2_SIF2_Drain_V
0xccb20000	HF_CH2_SIF2_Drain_C
0xccb40000	HF_CH2_SIF3_Drain_V
0xccb50000	HF_CH2_SIF3_Drain_C
0xcd1300e3	HF_CV1_DPFPP1
0xcda80000	HF_CV2_FIF1_Drain_V
0xcda90000	HF_CV2_FIF1_Drain_C
0xcdab0000	HF_CV2_FIF2_Drain_V
0xcdac0000	HF_CV2_FIF2_Drain_C
0xcdae0000	HF_CV2_SIF1_Drain_V
0xcdaf0000	HF_CV2_SIF1_Drain_C
0xcdb10000	HF_CV2_SIF2_Drain_V
0xcdb20000	HF_CV2_SIF2_Drain_C
0xcdb40000	HF_CV2_SIF3_Drain_V
0xcdb50000	HF_CV2_SIF3_Drain_C
0xcf110003	HF_CPR_CH_SLM
0xcf1300d2	HF_CPR_CHFPG1

0xcf140091 HF_CPR_CHFPG2
0xcf1500f9 HF_CPR_CHFPZ1
0xcf1600f3 HF_CPR_CHFPZ2
0xcf170088 HF_CPR_CHFPP2
0xcf180038 HF_CPR_CHFPG3
0xcf1900c7 HF_CPR_CHFPP3
0xcf260000 HF_CPR_Cal_Heater_C
0xcc2007ff HF_CH1_MXJNC_V
0xcc2207ff HF_CH1_MX_MG_C
0xcd2007ff HF_CV1_MXJNC_V
0xcd2207ff HF_CV1_MX_MG_C
0xcf1007ffHF_CPR_Chopper_Rot
0xcc1007ff HF_CH1_DPACT_C
0xcd1007ff HF_CV1_DPACT_C

HIFI_Configure_FCU_power:
0xcf080001 HF_CPR_Mixer_H_S
0xcf090001 HF_CPR_Mixer_V_S
0xcf0a0000 HF_CPR_Chopper_S
0xcf0b0001 HF_CPR_UCH_S
0xcf0c0001 HF_CPR_UCV_S

HIFI_Config_HRS_H_ATT_LO:
0xd5000000 HRH_switch
0xd580001f HRH_1U_ATT
0xd590001f HRH_1L_ATT
0xd5a0001f HRH_2U_ATT
0xd5b0001f HRH_2L_ATT
0xd5c0001f HRH_3U_ATT
0xd5d0001f HRH_3L_ATT
0xd5e0001f HRH_4U_ATT
0xd5f0001f HRH_4L_ATT
0xd5100030 HRH_Up_LO1
0xd5200030 HRH_Up_LO2
0xd5300030 HRH_Up_LO3
0xd5400030 HRH_Up_LO4
0xd5500030 HRH_Down_LO5
0xd5600030 HRH_Down_LO6
0xd6f00000 HRH_Down_LO7

HIFI_Config_HRS_V_Blocks:
0xd6008241 HRH_block1
0xd6108241 HRH_block2
0xd6208241 HRH_block3
0xd6308241 HRH_block4
0xd6408241 HRH_block5
0xd6508241 HRH_block6
0xd6608241 HRH_block7
0xd6708241 HRH_block8

HIFI_Config_HRS_V_ATT_LO:
0xd9000000 HRV_switch
0xd980001f HRV_1U_ATT
0xd990001f HRV_1L_ATT
0xd9a0001f HRV_2U_ATT
0xd9b0001f HRV_2L_ATT
0xd9c0001f HRV_3U_ATT
0xd9d0001f HRV_3L_ATT
0xd9e0001f HRV_4U_ATT
0xd9f0001f HRV_4L_ATT
0xd9100030 HRV_Up_LO1
0xd9200030 HRV_Up_LO2
0xd9300030 HRV_Up_LO3
0xd9400030 HRV_Up_LO4
0xd9500030 HRV_Down_LO5
0xd9600030 HRV_Down_LO6
0xdaf00000 HRV_Down_LO7

HIFI_Config_HRS_H_Blocks:
0xda008241 HRV_block1
0xda108241 HRV_block2
0xda208241 HRV_block3

0xda308241 HRV_block4
0xda408241 HRV_block5
0xda508241 HRV_block6
0xda608241 HRV_block7
0xda708241 HRV_block8

HIFI_Config_WBS_H:
0xe4000031 HWH_LASER_1_ON
0xe4000012 HWH_LASER_2_OFF
0xe4000014 HWH_Heater_0
0xe4000077 HWH_Latchup_low
0xe41fffef HWH_ATTENS(7,7,7,7,15)

HIFI_Config_WBS_V:
0xe8000031 HWV_LASER_1_ON
0xe8000012 HWV_LASER_2_OFF
0xe8000014 HWV_Heater_0
0xe8000077 HWV_Latchup_low
0xe81fffef HWV_ATTENS(7,7,7,7,15)

HIFI_Config_LCU:
0xf0001234 HL_RES_DROVR
0xf02117ff HL_CH1A_PLevel_V
0xf0212bbe HL_CH1A_M1_V
0xf02139bf HL_CH1A_M2_V
0xf02176fd HL_CH1A_Gate1_V
0xf02183f8 HL_CH1A_Gate2_V
0xf0219bff HL_CH1A_Drain1_V
0xf021a633 HL_CH1A_Drain2_V

HIFI_Config_LCU:
0xf0001234 HL_RES_DROVR
0xf02517ff HL_CH3A_PLevel_V
0xf0252c7e HL_CH3A_M1_V
0xf0253700 HL_CH3A_M2_V
0xf0254800 HL_CH3A_M3_V
0xf02572e6 HL_CH3A_Gate1_V
0xf02582f0 HL_CH3A_Gate2_V
0xf0259aff HL_CH3A_Drain1_V
0xf025a59b HL_CH3A_Drain2_V

HIFI_Single_cmd_simulator:
0xcc64c351 HIFI_Single_cmd
0xe4000009 HIFI_reset_WBS_H
0xe4000009 HIFI_reset_WBS_H
0xf000ffff HIFI_HL_Switch_off
0xf00ff0ff HIFI_HL_Standby
0xf00f0abc HIFI_HL_Nominal
0xf000cafe HIFI_HL_Reset
0xcc240000 HIFI_HF_CH1_DHTR_C
0xcd240000 HIFI_HF_CV1_DHTR_C
0xf0010110 HIFI_HL_switchon

A1.7 Spectroscopy Measurements

Table A1.7.1 Total Power measurements expected results:

Meas. ID	WBS H+V n. seq.	WBS H+V SD Pack.	WBS H+V SD start	WBS H+V IF Power	HRS H+V n. seq.	HRS H+V SD Pack.	HRS H+V SD start	HRS H+V IF Power	WBS H Ref file	WBS V Ref file	HRS H Ref file	HRS V Ref file
1	2	56	2	2	2	32	2	2	406_reference	407_reference	404_reference	405_reference
2	2	56	2	2	2	32	2	2	406_coad1_rs1	407_coad1_rs1	404_coad1_rs1	405_coad1_rs1
3	2	56	2	2	2	32	2	2	406_coad1_rs2	407_coad1_rs2	404_coad1_rs2	405_coad1_rs2
4	2	56	2	2	2	32	2	2	406_coad1_rs16	407_coad1_rs16	404_coad1_rs16	405_coad1_rs16

5	2	56	2	2	2	32	2	2	406_coad2_rs0	407_coad2_rs0	404_coad2_rs0	405_coad2_rs0
6	2	56	2	2	2	32	2	2	406_coad2_rs1	407_coad2_rs1	404_coad2_rs1	405_coad2_rs1
7	2	56	2	2	2	32	2	2	406_coad4_rs0	407_coad4_rs0	404_coad4_rs0	405_coad4_rs0
8	2	56	2	2	2	32	2	2	406_coad4_rs2	407_coad4_rs2	404_coad4_rs2	405_coad4_rs2
10	2	14	2	2	2	6	2	2	406_CCD1	407_CCD2	404_range160	405_range10
11	2	32	2	2	2	14	2	2	406_mixrange	407_mixrange	404_range170	405_range85
12	2	56	2	2	2	32	2	2	406_coad4_rs0	407_coad4_rs0	404_coad32_rs0	405_coad32_rs0

Table A1.7.2 Total Power measurements expected IF Powers:

Meas. ID	WBS H ifpower				WBS V ifpower				HRS H ifpower								HRS V ifpower							
1	290	814	270	272	288	673	268	271	499	500	500	499	499	500	500	500	13941	13941	13941	13941	13941	13941	13941	13941
2	145	407	135	1	144	336	134	1	499	500	500	499	499	500	500	500	13941	13941	13941	13941	13941	13941	13941	13941
3	72	203	67	68	72	168	67	67	499	500	500	499	499	500	500	500	13941	13941	13941	13941	13941	13941	13941	13941
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16333	16333	16333	16333	16333	16333	16333	16333
5	145	407	135	1	144	336	134	1	249	250	250	249	250	250	250	250	6970	6970	6970	6970	6970	6970	6970	6970
6	72	203	67	68	72	168	67	67	249	250	250	249	250	250	250	250	6970	6970	6970	6970	6970	6970	6970	6970
7	72	203	67	68	72	168	67	67	124	125	125	124	125	125	125	125	3485	3485	3485	3485	3485	3485	3485	3485
8	18	50	16	17	18	42	16	16	124	125	125	124	125	125	125	125	3485	3485	3485	3485	3485	3485	3485	3485
10	18	50	16	17	18	42	16	16	124	125	125	124	125	125	125	125	3485	3485	3485	3485	3485	3485	3485	3485
11	18	50	16	17	18	42	16	16	124	125	125	124	125	125	125	125	3485	3485	3485	3485	3485	3485	3485	3485
12	72	203	67	68	72	168	67	67	15	15	15	15	15	15	15	15	435	435	435	435	435	435	435	435

Table A1.7.3 Fast Chop measurements expected results:

Meas. ID	WBS H+V n. seq.	WBS H+V SD Pack.	WBS H+V SD start	WBS H+V IF Power	HRS H+V n. seq.	HRS H+V SD Pack.	HRS H+V SD start	HRS H+V IF Power	WBS H Ref file	WBS V Ref file	HRS H Ref file	HRS V Ref file
1	4	56*2	4	4	20	16*20	20	20	406_FC1_A_ref 406_FC1_B_ref	407_FC1_A_ref 407_FC1_B_ref	404_FC1_A_ref 404_FC1_B_ref	405_FC1_A_ref 405_FC1_B_ref
2	4	56*2	4	4	24	16*24	24	24	406_FC1_A_ref 406_FC1_B_ref	407_FC1_A_ref 407_FC1_B_ref	404_FC1_A_ref 404_FC1_B_ref	405_FC1_A_ref 405_FC1_B_ref

Table A1.7.3 Fast Chop expected IF powers:

TBW

A1.8 HRS Tune

Test case	HRS H ref files	HRS V ref files	HRS H Expected Att. Setting								HRS V Expected Att. Setting							
Nominal	HRS_tune_nom_1	HRS_tune_nom_1	26	10	10	10	10	10	26	26	26	10	10	10	10	10	26	26
	HRS_tune_nom_2	HRS_tune_nom_2	26	6	9	10	10	10	26	26	26	6	9	10	10	10	26	26
Low signal	HRS_tune_LS_1	HRS_tune_LS_1	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
	HRS_tune_LS_2	HRS_tune_LS_2	7	8	10	6	8	11	9	9	7	8	10	6	8	11	9	9

A1.9 WBS Tune

TBW

Test case	WBS H ref files	WBS V ref files	WBS H Expected Att. Setting								WBS V Expected Att. Setting							

A1.10 Mixer Magnet Tune

TBW

Test case	H ref files	V ref files																
Tune use HRS																		
Tune use WBS																		

List of expected cmd and hk requests issued by OBS. To be written

A1.11 WBS COMB

List of expected cmd and hk requests issued by OBS and related timing constraints.
To be written

A2. Appendix CDMS Local Commands

A2.1 Telecommand Acceptance local commands

TC file name	TC file content:	Description
H_conn_test.txt (TC 17.1)	1C00 e80e 0005 0111 0100 dc3b	Perform Connection Test Command
APID_test.txt	1b00 e80e 0005 0111 0100 c0c1	Perform Connection Test Command, with an incorrect APID of 0x300
CRC_test.txt	1C00 e80e 0005 0111 0100 1111	Same as TC17.1, but with an incorrect checksum of 0x1111
Length_test.txt	1C00 e80e 000A 0111 0100 b9c2	Same as TC17.1, but with an incorrect packet length of 0xA
Type_test.txt	1C00 e80e 0005 0101 0100 9f58	Same as TC17.1, but with an incorrect packet type of 0x1
Subtype_test.txt	1C00 e80e 0005 0111 0a00 00c1	Same as TC17.1, but with an incorrect packet subtype of 0xA
Ack0_test.txt	1C00 e80e 0005 0011 0100 aa8f	Same as TC17.1, but with the "ack" bits in the TC header set to '0000'
Ack3_test.txt	1C00 e80e 0005 0311 0100	Same as TC17.1, but with the "ack" bits in the TC header set to '0011'
Ack5_test.txt	1C00 e80e 0005 0511 0100 f	Same as TC17.1, but with the "ack" bits in the TC header set to '0101'
Ack9_test.txt	1C00 e80e 0005 0911 0100	Same as TC17.1, but with the "ack" bits in the TC header set to '1001'

A2.2 TP4 –Memory management Commands

Recall that the memory commands work on a memory area that is specified by an ID, an offset and a length (number of words). The ID specifies the type of memory (e.g. PM, DM, dual port RAM) of a memory segment and a starting address of the segment. The offset amd number of words identify the start and end part of the memory over which the command is acting.
In order to carry out the memory service testing the following memory areas will be used:

Mnemonic	Offset	Length (SAU)	ID	Notes
DM	0x22000	0x4	1	Short segment (4cells x 4byte = 16 bytes) of the data memory. Out of the memory used by the OBS.
PM	0x60000	0x4	0	Short segment (4cells x 6bytes = 24 bytes) of the data memory. Out of the PM memory used by the OBS.
DML	0x22000	0xFFFF	1	Long segment of the data memory. Out of the memory used by the OBS.
PML	0x5000	0xFFFF	0	Long segment of the PM memory. This segment contains part of the OBS code.
EEP	0x0000	0x4	3	Short segment (4cells x 4byte = 16 bytes) of the EEPROM. This segment is write only.

The EEP segment is included for future use but is not employed in the current test procedure.
The following commands are employed in the test.

Command TP4.1
A mem_load command that loads the 4 (48bits) words of the PM segment with the following values: 0xAAAABBBBCCCC. This command will be reffered as **mem_load_PM**. The following table reports the command fields values.

Position	Length(bits)	Field	Value
10	8	Memory ID	0x0
11	24	Start address	0x60000
14	8	Spare	0
15	8	Length	0x4
16	16		0xAAAA
18	16		0xBBBB
20	16		0xCCCC
22	16		0xAAAA
24	16		0xBBBB

26	16		0xCCCC
28	16		0xAAAA
30	16		0xBBBB
32	16		0xCCCC
34	16		0xAAAA
36	16		0xBBBB
38	16		0xCCCC
40	16	Data checksum	0xb3ba

The command is stored in the text file mem_load_PM.txt and it has the following hex value:
1C00 D215 0025 0f06 0200 0006 0000 0004 AAAA BBBB CCCC AAAA BBBB CCCC AAAA BBBB CCCC AAAA BBBB CCCC b3ba dd79
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet header which is not reported in the table.
Command TP4.2
A mem_dump command that dumps the PM segment. This command will be reffered as **mem_dump_PM**. The following table reports the command fields values.

Position	Length	Field	Value
10	8	Memory ID	0x0
11	24	Start address	0x60000
14	8	Spare	0
15	8	Length	0x4

The command is stored in the text file mem_dump_PM.txt and it has the following hex value:
1C00 D215 000b 0f06 0500 0006 0000 0004 d12b
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet header which is not reported in the table.

Command TP4.3
A mem_check command that computes the CRC over the PM segment. This command will be reffered as **mem_check_PM**. The following table reports the command fields values.

Position	Length	Field	Value
10	8	Memory ID	0x0
11	24	Start address	0x60000
14	8	Spare	0
15	8	Length	0x4

The command is stored in the text file mem_check_PM.txt and it has the following hex value:
1C00 D215 000b 0f06 0900 0006 0000 0004 c09c
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet header which is not reported in the table.

Command TP4.4
A mem_load command that loads the 4 (32bits) words of the DM segment with the following values: 0xAAAABBBB. This command will be reffered as **mem_load_DM**. The following table reports the command fields values.

Position	Length	Field	Value
10	8	Memory ID	0x1
11	24	Start address	0x22000
14	8	Spare	0
15	8	Length	0x4
16	16	Data	0xAAAA
18	16	Data	0xBBBB
20	16	Data	0xAAAA
22	16	Data	0xBBBB
24	16	Data	0xAAAA
26	16	Data	0xBBBB
28	16	Data	0xAAAA
30	16	Data	0xBBBB
32	16	Data checksum	0x446

The command is stored in the text file mem_load_DM.txt and it has the following hex value:
1C00 D215 001d 0f06 0200 0102 2000 0004 AAAA BBBB AAAA BBBB AAAA BBBB AAAA BBBB 0446 80af
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet header which is not reported in the table.

Command TP4.5
A mem_dump command that dumps the DM segment. This command will be reffered as **mem_dump_DM**. The following table reports the command fields values.

Position	Length	Field	Value
10	8	Memory ID	0x1
11	24	Start address	0x22000
14	8	Spare	0
15	8	Length	0x4

The command is stored in the text file mem_dump_DM.txt and it has the following hex value:
1C00 D215 000b 0f06 0500 0102 2000 0004 2ac3
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet header which is not reported in the table.

Command TP4.6
A mem_check command that computes the CRC over the DM segment. This command will be reffered as **mem_check_DM**. The following table reports the command fields values.

Position	Length	Field	Value
10	8	Memory ID	0x1
11	24	Start address	0x22000
14	8	Spare	0
15	8	Length	0x4

The command is stored in the text file mem_check_DM.txt and it has the following hex value:
1C00 D215 000b 0f06 0900 0102 2000 0004 3b74
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet header which is not reported in the table.

Command TP4..7
A mem_dump command that dumps the DML segment. This command will be reffered as **mem_dump_long_DM**. The following table reports the command fields values.

Position	Length	Field	Value
10	8	Memory ID	0x1
11	24	Start address	0x22000
14	8	Spare	0
15	8	Length	0xFFFF

The command is stored in the text file mem_dump_long_DM.txt and it has the following hex value:
1C00 D215 000b 0f06 0500 0102 2000 ffff 7748
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet header which is not reported in the table.
Command TP4.8
A mem_dump command that dumps the PML segment. This command will be reffered as **mem_dump_long_PM**. The following table reports the command fields values.

Position	Length	Field	Value
10	8	Memory ID	0x1
11	24	Start address	0x22000
14	8	Spare	0
15	8	Length	0xFFFF

The command is stored in the text file mem_dump_long_DM.txt and it has the following hex value:
1C00 D215 000b 0f06 0500 0000 5000 ffff 341e
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet header which is not reported in the table.

Command TP4.9
A mem_dump_abort command that stops any dumping activity. This command will be reffered as **mem_dump_abort**. The command has no parameters. The command is stored in the text file mem_dump_abort.txt and it has the following hex value:
1C00 D215 0005 0f06 0b00 fcad
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet

header.

Command TP4.1	**mem_load_PM**.
Command TP4.2	**mem_dump_PM**.
Command TP4.3	**mem_check_PM**.
Command TP4.4	**mem_load_DM**.
Command TP4.5	**mem_dump_DM**.
Command TP4.6	**mem_check_DM**.
Command TP4..7	**mem_dump_long_DM**.
Command TP4.8	**mem_dump_long_PM**.
Command TP4.9	**mem_dump_abort**.

A2.3 Total Power tests

Table A2.3.1 TP tests configuration parameters:

Meas. ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1	1	1	1	1	5	5	1005	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	0	0	255	255	24
2	1	1	1	1	1	5	5	1005	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	1	1	255	255	24
3	1	1	1	1	1	5	5	1005	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	2	2	255	255	24
4	1	1	1	1	1	5	5	1005	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	16	16	255	255	24
5	1	2	1	2	2	5	5	1005	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	0	0	255	255	24
6	1	2	1	2	2	5	5	1005	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	1	1	255	255	24
7	1	4	1	4	4	5	5	1005	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	0	0	255	255	24
8	1	4	1	4	4	5	5	1005	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	2	2	255	255	24
9	1	4	1	4	4	5	5	1005	100	0	0	2048	0	4096	0	6144	0	0	0	2048	0	4096	0	6144	0	2	2	0	0	24
10	1	4	1	4	4	5	5	1005	100	0	2048	2048	0	4096	0	6144	0	0	0	2048	2048	4096	0	6144	0	2	2	160	10	24
11	1	4	1	4	4	5	5	1005	100	0	2048	2048	2048	5120	2048	6144	2048	0	100	2048	100	4096	100	6144	100	2	2	170	85	24
12	2	4	8	4	32	5	5	1205	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	0	0	255	255	24

Table A2.3.2 FC tests configuration parameters:

Meas. ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1	1	1	1	4	5	5	2005	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	0	0	255	255	24
2	1	4	1	4	4	5	5	2005	100	0	2048	2048	2048	4096	2048	6144	2048	0	2048	2048	2048	4096	2048	6144	2048	2	2	255	255	24

Meas. ID	HIF_CPR_CH_ROT1	HIF_CPR_CH_ROT1	HIF_N_WBS1	HIF_N_HRS_TRANS
1	0	0	2	4
2	0	0	4	4

Table A2.3.3 Column Identifiers for the Spectroscopy measurements configuration parameters

HIFI_BB_ID	1
HIF_N_WBS_START	2
HIF_R_HRS	3
HIF_N_WBS_INTEGR	4
HIF_N_HRS_INTEGR	5
HIF_DEL_HRS	6
HIF_DEL_WBS	7
HIF_T_ACC_WBS	8
HIF_T_ACC_HRS	9
HIF_WBSH_OFFSET1	10
HIF_WBSH_WIDTH1	11
HIF_WBSH_OFFSET2	12
HIF_WBSH_WIDTH2	13
HIF_WBSH_OFFSET3	14
HIF_WBSH_WIDTH3	15
HIF_WBSH_OFFSET4	16
HIF_WBSH_WIDTH4	17
HIF_WBSV_OFFSET1	18
HIF_WBSV_WIDTH1	19
HIF_WBSV_OFFSET2	20
HIF_WBSV_WIDTH2	21
HIF_WBSV_OFFSET3	22
HIF_WBSV_WIDTH3	23
HIF_WBSV_OFFSET4	24
HIF_WBSV_WIDTH4	25
HIF_HRS_RSHIFT	26
HIF_WBS_RSHIFT	27
HIF_HRSH_SEL	28
HIF_HRSV_SEL	29
HIF_WBS_packing	30

A2.4 TP6 Configure Spectroscopy errors

conf_spect_new_err_1.

The command is stored in the text file conf_spect_new_err_1.txt and it has the following hex value:
1C00 D205 0045 0908 0400 0B11 0000 0000 0000 0001 0001 0001 0001 000E 000E 07D5 03E8 0000 0800 0800 0800 1000 0800 1800 0800 0000 0800 0800 0800 1000 0800 1800 0800 0000 0000 00FF 00FF 12AB
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet header which is not reported in the table.

conf_spect_new_err_2

The WBS integration time is set to 1024 (hex 0800) which is not a multiple of 10 plus 5 as it should be (see AD1). The wrong command will be refferred as.

The command is stored in the text file conf_spect_new_err_2.txt and it has the following hex value:
1C00 D205 0047 0908 0400 0B11 0000 0000 0000 0001 0001 0001 0001 000E 000E 0800 03E8 0000 0800 0800 0800 1000 0800 1800 0800 0000 0800 0800 0800 1000 0800 1800 0800 0000 0000 00FF 00FF 0001 991E
where the last four hex digits are the CRC, and the first 5x4 hex digits are the source packet header which is not reported in the table.

A2.5
A3. TOPE scripts

A3.1 Initope

SCRIPT:

```
source ../TC/readCCF.incl
source ../TC/readCPC.incl
# source ../TC/readPCF.incl


------------------------------------------------------------------------
INCLUDE Files:

readCCF.incl

set ch [open "~/data/ASCII/ccf.dat" r]

while {! [eof $ch] } {
  gets $ch str
  scan $str %s%s cname descr
  eval [ concat "set "  $descr $cname]
}
close $ch

------------------------------------------------------------------------
readCPC.incl

set ch [open "~/data/ASCII/cpc.dat" r]

while {! [eof $ch] } {
  gets $ch str
  scan $str %s%s pname descr
  eval [ concat "set " $descr $pname]
}
close $ch
```

A3.2 LimitCheck

```
# send value

tcsend $HIFI_H_DHTR_C_check_on " $HIF_N_breach 5 " " $HIF_H_DHTR_Max_C 50001 "
tcsend $HIFI_V_DHTR_C_check_on " $HIF_N_breach 5 " " $HIF_V_DHTR_Max_C 50001 "

tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3429105665"
tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3445882881"

waittime 7

tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3429155665"
tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3445932881"

waittime 7

tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3429155666"
tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3445932882"
```

```
waittime 10

tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3429155665"
tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3445932881"

waittime 7

tcsend $HIFI_H_DHTR_C_check_off
tcsend $HIFI_V_DHTR_C_check_off

tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3429155666"
tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3445932882"

waittime 10

tcsend $HIFI_WH_Laser_T_check_on " $HIF_N_breach 15 " " $HIF_HWH_laser_Max_T 398 "
tcsend $HIFI_WV_Laser_T_check_on " $HIF_N_breach 15 " " $HIF_HWV_laser_Max_T 398 "

waittime 20

tcsend $HIFI_WH_Laser_T_check_off
tcsend $HIFI_WV_Laser_T_check_off

waittime 10

tcsend $HIFI_WH_Laser_T_check_on " $HIF_N_breach 5 " " $HIF_HWH_laser_Max_T 405 "
tcsend $HIFI_WV_Laser_T_check_on " $HIF_N_breach 5 " " $HIF_HWV_laser_Max_T 405 "

waittime 10

tcsend $HIFI_WH_Laser_T_check_off
tcsend $HIFI_WV_Laser_T_check_off

waittime 10

tcsend $HIFI_WH_Laser_T_check_on " $HIF_N_breach 5 " " $HIF_HWH_laser_Max_T 0 "
tcsend $HIFI_WV_Laser_T_check_on " $HIF_N_breach 5 " " $HIF_HWV_laser_Max_T 0 "

waittime 10

tcsend $HIFI_WH_Laser_T_check_off
tcsend $HIFI_WV_Laser_T_check_off

waittime 10

tcsend $HIFI_FCU_nonresp_check_on " $HIF_N_breach_8C 5 " " $HIF_N_breach_8D 10 " " $HIF_N_breach_8F
15 "

waittime 20

tcsend $HIFI_FCU_nonresp_check_off
```

A3.3 configSubsystems.tcl

```
source ../TC/HIFI_Configure_FCU.incl

source ../TC/HIFI_Configure_FCU_Power.incl
source ../TC/HIFI_Config_HRS_H_att_lo.incl
source ../TC/HIFI_Config_HRS_H_blocks.incl
source ../TC/HIFI_Config_HRS_V_att_lo_1.incl
source ../TC/HIFI_Config_HRS_V_blocks_1.incl
source ../TC/HIFI_Configure_WBS_H.incl
source ../TC/HIFI_Configure_WBS_V.incl
source ../TC/HIFI_Configure_LCU_ch1a.incl
source ../TC/HIFI_Configure_LCU_ch3a.incl
#source ../TC/HIFI_Configure_LCU_ch3b.incl
#source ../TC/HIFI_Configure_LCU_ch6a.incl
```

Used include files:

```
HIFI_Configure_FCU.incl:

tcsend $HIFI_Configure_FCU  ack {ACCEPT}\
   " $HIFI_BB_ID       0       RAW " \
   " $HF_CPR_MXBAND    1       RAW " \
   " $HF_CH1_DPFPP1    227     RAW " \
```

```
   " $HF_CH2_FIF1_Drain_V     0       ENG " \
   " $HF_CH2_FIF1_Drain_C     0       ENG " \
   " $HF_CH2_FIF2_Drain_V     0       ENG " \
   " $HF_CH2_FIF2_Drain_C     0       ENG " \
   " $HF_CH2_SIF1_Drain_V     0       ENG " \
   " $HF_CH2_SIF1_Drain_C     0       ENG " \
   " $HF_CH2_SIF2_Drain_V     0       ENG " \
   " $HF_CH2_SIF2_Drain_C     0       ENG " \
   " $HF_CH2_SIF3_Drain_V     0       ENG " \
   " $HF_CH2_SIF3_Drain_C     0       ENG " \
   " $HF_CV1_DPFPP1    227     RAW " \
   " $HF_CV2_FIF1_Drain_V     0       ENG " \
   " $HF_CV2_FIF1_Drain_C     0       ENG " \
   " $HF_CV2_FIF2_Drain_V     0       ENG " \
   " $HF_CV2_FIF2_Drain_C     0       ENG " \
   " $HF_CV2_SIF1_Drain_V     0       ENG " \
   " $HF_CV2_SIF1_Drain_C     0       ENG " \
   " $HF_CV2_SIF2_Drain_V     0       ENG " \
   " $HF_CV2_SIF2_Drain_C     0       ENG " \
   " $HF_CV2_SIF3_Drain_V     0       ENG " \
   " $HF_CV2_SIF3_Drain_C     0       ENG " \
   " $HF_CPR_CH_SINE_S ON      ENG " \
   " $HF_CPR_CH_LOOP_S CLOSE   ENG " \
   " $HF_CPR_CHFPG1    210     RAW " \
   " $HF_CPR_CHFPG2    145     RAW " \
   " $HF_CPR_CHFPZ1    249     RAW " \
   " $HF_CPR_CHFPZ2    243     RAW " \
   " $HF_CPR_CHFPP2    136     RAW " \
   " $HF_CPR_CHFPG3    56      RAW " \
   " $HF_CPR_CHFPP3    199     RAW " \
   " $HF_CPR_Cal_Heater_C     0       ENG " \
   " $HF_CH1_MXBIAS_V  0       ENG " \
   " $HF_CH1_MX_MG_C   0       ENG " \
   " $HF_CV1_MXBIAS_V  0       ENG " \
   " $HF_CV1_MX_MG_C   0       ENG " \
   " $HF_CPR_Chopper_Rot      0       ENG " \
   " $HF_CH1_DPACT_C   0       ENG " \
   " $HF_CV1_DPACT_C   0       ENG " \
   referby HIFI_Configure_FCU_flg

-------------------------------------------------------------------------------
HIFI_Config_HRS_H_att_lo.incl

tcsend $HIFI_Config_HRS_H_att_lo  ack {ACCEPT}\
   " $HIFI_BB_ID         0         " \
   " $HRH_switch         H       ENG " \
   " $HRH_1U_ATT         15.5    ENG " \
   " $HRH_1L_ATT         15.5    ENG " \
   " $HRH_2U_ATT         15.5    ENG " \
   " $HRH_2L_ATT         15.5    ENG " \
   " $HRH_3U_ATT         15.5    ENG " \
   " $HRH_3L_ATT         15.5    ENG " \
   " $HRH_4U_ATT         15.5    ENG " \
   " $HRH_4L_ATT         15.5    ENG " \
   " $HRH_Up_OL1_M       0         " \
   " $HRH_Up_OL1_A       0         " \
   " $HRH_Up_OL2_M       0         " \
   " $HRH_Up_OL2_A       0         " \
   " $HRH_Up_OL3_M       0         " \
   " $HRH_Up_OL3_A       0         " \
   " $HRH_Up_OL4_M       0         " \
   " $HRH_Up_OL4_A       0         " \
   " $HRH_Down_OL5_M     0         " \
   " $HRH_Down_OL5_A     0         " \
   " $HRH_Down_OL6_M     0         " \
   " $HRH_Down_OL6_A     0         " \
   " $HRH_Down_OL7_M     0         " \
   referby HIFI_Config_HRS_H_att_lo_flg

tcsend $HIFI_Conf_3_HRS_H_att_lo  ack {ACCEPT}\
   " $HIFI_BB_ID        0         " \
   " $HRH_switch        H       ENG " \
   " $HRH_1U_ATT        15.5    ENG " \
   " $HRH_1L_ATT        15.5    ENG " \
   " $HRH_2U_ATT        15.5    ENG " \
   " $HRH_2L_ATT        15.5    ENG " \
   " $HRH_3U_ATT        15.5    ENG " \
   " $HRH_3L_ATT        15.5    ENG " \
   " $HRH_4U_ATT        15.5    ENG " \
   " $HRH_4L_ATT        15.5    ENG " \
   " $HRH_Up_OL1_M      0         " \
   " $HRH_Up_OL1_A      0         " \
   " $HRH_Up_OL2_M      0         " \
```

```
   " $HRH_Up_OL2_A     0        " \
   " $HRH_Up_OL3_M     0        " \
   " $HRH_Up_OL3_A     0        " \
   " $HRH_Up_OL4_M     0        " \
   " $HRH_Up_OL4_A     0        " \
   " $HRH_Down_OL5_M   0        " \
   " $HRH_Down_OL5_A   0        " \
   " $HRH_Down_OL6_M   0        " \
   " $HRH_Down_OL6_A   0        " \
   " $HRH_Down_OL7_M   0        " \
   referby HIFI_Config_HRS_H_att_lo_flg

------------------------------------------------------------------------------
HIFI_Config_HRS_H_blocks.incl

tcsend $HIFI_Config_HRS_H_blocks  ack {ACCEPT}\
   " $HIFI_BB_ID        0        " \
   " $HRH_Block_1       sine_ultra      ENG " \
   " $HRH_Block_2       sine_ultra      ENG " \
   " $HRH_Block_3       sine_ultra      ENG " \
   " $HRH_Block_4       sine_ultra      ENG " \
   " $HRH_Block_5       sine_ultra      ENG " \
   " $HRH_Block_6       sine_ultra      ENG " \
   " $HRH_Block_7       sine_ultra      ENG " \
   " $HRH_Block_8       sine_ultra      ENG " \
   referby HIFI_Config_HRS_H_blocks_flg


tcsend $HIFI_Conf_3_HRS_H_blocks  ack {ACCEPT}\
   " $HIFI_BB_ID       0        " \
   " $HRH_Block_1      sine_ultra     ENG " \
   " $HRH_Block_2      sine_ultra     ENG " \
   " $HRH_Block_3      sine_ultra     ENG " \
   " $HRH_Block_4      sine_ultra     ENG " \
   " $HRH_Block_5      sine_ultra     ENG " \
   " $HRH_Block_6      sine_ultra     ENG " \
   " $HRH_Block_7      sine_ultra     ENG " \
   " $HRH_Block_8      sine_ultra     ENG " \
   referby HIFI_Config_HRS_H_blocks_flg

------------------------------------------------------------------------------
HIFI_Config_HRS_V_att_lo_1.incl

tcsend $HIFI_Config_HRS_V_att_lo  ack {ACCEPT}\
   " $HIFI_BB_ID       0        " \
   " $HRV_switch       V       ENG " \
   " $HRV_1U_ATT       15.5    ENG " \
   " $HRV_1L_ATT       15.5    ENG " \
   " $HRV_2U_ATT       15.5    ENG " \
   " $HRV_2L_ATT       15.5    ENG " \
   " $HRV_3U_ATT       15.5    ENG " \
   " $HRV_3L_ATT       15.5    ENG " \
   " $HRV_4U_ATT       15.5    ENG " \
   " $HRV_4L_ATT       15.5    ENG " \
   " $HRV_Up_OL1_M     0        " \
   " $HRV_Up_OL1_A     0        " \
   " $HRV_Up_OL2_M     0        " \
   " $HRV_Up_OL2_A     0        " \
   " $HRV_Up_OL3_M     0        " \
   " $HRV_Up_OL3_A     0        " \
   " $HRV_Up_OL4_M     0        " \
   " $HRV_Up_OL4_A     0        " \
   " $HRV_Down_OL5_M   0        " \
   " $HRV_Down_OL5_A   0        " \
   " $HRV_Down_OL6_M   0        " \
   " $HRV_Down_OL6_A   0        " \
   " $HRV_Down_OL7_M   0        " \
   referby HIFI_Config_HRS_V_att_lo_flg

------------------------------------------------------------------------------
HIFI_Config_HRS_V_blocks_1.incl

tcsend $HIFI_Config_HRS_V_blocks  ack {ACCEPT}\
   " $HIFI_BB_ID       0        " \
   " $HRV_Block_1      sine_ultra     ENG " \
   " $HRV_Block_2      sine_ultra     ENG " \
   " $HRV_Block_3      sine_ultra     ENG " \
   " $HRV_Block_4      sine_ultra     ENG " \
   " $HRV_Block_5      sine_ultra     ENG " \
   " $HRV_Block_6      sine_ultra     ENG " \
   " $HRV_Block_7      sine_ultra     ENG " \
   " $HRV_Block_8      sine_ultra     ENG " \
```

```
   referby HIFI_Config_HRS_V_blocks_flg

------------------------------------------------------------------------------
HIFI_Configure_WBS_H.incl


tcsend $HIFI_Configure_WBS_H  ack {ACCEPT}\
   " $HIFI_BB_ID         0         " \
   " $HWH_LASER1_S       ON       ENG " \
   " $HWH_LASER2_S       OFF      ENG " \
   " $HWH_Heater         0         " \
   " $HWH_Latchup_S      Level2   ENG " \
   " $HWH_ATT_Band_4     7         " \
   " $HWH_ATT_Band_3     7         " \
   " $HWH_ATT_Band_2     7         " \
   " $HWH_ATT_Band_1     7         " \
   " $HWH_ATT_IN         15        " \
   referby HIFI_Configure_WBS_H_flg


tcsend $HIFI_Conf_3ure_WBS_H  ack {ACCEPT}\
   " $HIFI_BB_ID       0         " \
   " $HWH_LASER1_S     ON       ENG " \
   " $HWH_LASER2_S     OFF      ENG " \
   " $HWH_Heater       0         " \
   " $HWH_Latchup_S    Level2   ENG " \
   " $HWH_ATT_Band_4   7         " \
   " $HWH_ATT_Band_3   7         " \
   " $HWH_ATT_Band_2   7         " \
   " $HWH_ATT_Band_1   7         " \
   " $HWH_ATT_IN       15        " \
   referby HIFI_Configure_WBS_H_flg


------------------------------------------------------------------------------
HIFI_Configure_WBS_V.incl

tcsend $HIFI_Configure_WBS_V  ack {ACCEPT}\
   " $HIFI_BB_ID         0         " \
   " $HWV_LASER1_S       ON       ENG " \
   " $HWV_LASER2_S       OFF      ENG " \
   " $HWV_Heater         0         " \
   " $HWV_Latchup_S      Level2   ENG " \
   " $HWV_ATT_Band_4     7         " \
   " $HWV_ATT_Band_3     7         " \
   " $HWV_ATT_Band_2     7         " \
   " $HWV_ATT_Band_1     7         " \
   " $HWV_ATT_IN         15        " \
   referby HIFI_Configure_WBS_V_flg

tcsend $HIFI_Conf_3ure_WBS_V  ack {ACCEPT}\
   " $HIFI_BB_ID       0         " \
   " $HWV_LASER1_S     ON       ENG " \
   " $HWV_LASER2_S     OFF      ENG " \
   " $HWV_Heater       0         " \
   " $HWV_Latchup_S    Level2   ENG " \
   " $HWV_ATT_Band_4   7         " \
   " $HWV_ATT_Band_3   7         " \
   " $HWV_ATT_Band_2   7         " \
   " $HWV_ATT_Band_1   7         " \
   " $HWV_ATT_IN       15        " \
   referby HIFI_Configure_WBS_V_flg

------------------------------------------------------------------------------
HIFI_Configure_LCU_ch1a.incl

tcsend $HIFI_Configure_LCU_ch1a  ack {ACCEPT}\
   " $HIFI_BB_ID         0         " \
   " $HL_PLEVEL_C        0.21     ENG " \
   " $HL_M1_1A_V         7.5      ENG " \
   " $HL_M2_1A_V         3.5      ENG " \
   " $HL_Gate1_1A_V      -0.5     ENG " \
   " $HL_Gate2_1A_V      -2       ENG " \
   " $HL_Drain1_1A_V     3        ENG " \
   " $HL_Curlim1         1.22     ENG " \
   " $HL_Drain2_1A_V     1.55     ENG " \
   " $HL_Curlim2         1.22     ENG " \
   referby HIFI_Configure_LCU_ch1a_flgO


tcsend $HIFI_Conf_3ure_LCU_ch1a  ack {ACCEPT}\
   " $HIFI_BB_ID       0         " \
   " $HL_PLEVEL_C      0.21     ENG " \
```

```
   " $HL_M1_1A_V       7.5     ENG " \
   " $HL_M2_1A_V       3.5     ENG " \
   " $HL_Gate1_1A_V    -0.5    ENG " \
   " $HL_Gate2_1A_V    -2      ENG " \
   " $HL_Drain1_1A_V   3       ENG " \
   " $HL_Curlim1       1.22    ENG " \
   " $HL_Drain2_1A_V   1.55    ENG " \
   " $HL_Curlim2       1.22    ENG " \
   referby HIFI_Configure_LCU_ch1a_flg


------------------------------------------------------------------------------
HIFI_Configure_LCU_ch3a.incl

tcsend $HIFI_Configure_LCU_ch3a  ack {ACCEPT}\
   " $HIFI_BB_ID       0        " \
   " $HL_PLEVEL_C      0.21    ENG " \
   " $HL_M1_3A_V       9       ENG " \
   " $HL_M2_3A_V       -2      ENG " \
   " $HL_M3_3A_V       0       ENG " \
   " $HL_Gate1_3A_V    -2.53   ENG " \
   " $HL_Gate2_3A_V    -2.51   ENG " \
   " $HL_Drain1_3A_V   2.75    ENG " \
   " $HL_Curlim1       1.22    ENG " \
   " $HL_Drain2_3A_V   1.4     ENG " \
   " $HL_Curlim2       1.22    ENG " \
   referby HIFI_Configure_LCU_ch3a_flg


tcsend $HIFI_Conf_3ure_LCU_ch3a  ack {ACCEPT}\
   " $HIFI_BB_ID        0        " \
   " $HL_PLEVEL_C       0.21    ENG " \
   " $HL_M1_3A_V        9       ENG " \
   " $HL_M2_3A_V        -2      ENG " \
   " $HL_M3_3A_V        0       ENG " \
   " $HL_Gate1_3A_V     -2.53   ENG " \
   " $HL_Gate2_3A_V     -2.51   ENG " \
   " $HL_Drain1_3A_V    2.75    ENG " \
   " $HL_Curlim1        1.22    ENG " \
   " $HL_Drain2_3A_V    1.4     ENG " \
   " $HL_Curlim2        1.22    ENG " \
   referby HIFI_Configure_LCU_ch3a_flg
```

A3.4 HIFI_Single_cmd_simulator.tcl

```
tcsend $HIFI_Single_cmd  " $HIFI_BB_ID 0 " " $HIFI_cmd 3429155665"
tcsend $HIFI_Reset_WBS_H  " $HIFI_BB_ID 1 " " $HIFI_cmd 3429155666"
tcsend $HIFI_Reset_WBS_H  " $HIFI_BB_ID 2 " " $HIFI_cmd 3429155667"
tcsend $HIFI_HL_Switch_off  " $HIFI_BB_ID 3 " " $HIFI_cmd 3429155668"
tcsend $HIFI_HL_Standby  " $HIFI_BB_ID 4 " " $HIFI_cmd 3429155669"
tcsend $HIFI_HL_Nominal  " $HIFI_BB_ID 5 " " $HIFI_cmd 3429155670"
tcsend $HIFI_HL_Reset  " $HIFI_BB_ID 6 " " $HIFI_cmd 3429155671"
tcsend $HIFI_set_HF_CH1_DHTR_C  " $HIFI_BB_ID 7 " " $HIFI_cmd 3429155672"
tcsend $HIFI_set_HF_CV1_DHTR_C  " $HIFI_BB_ID 8 " " $HIFI_cmd 3429155673"
tcsend $HIFI_HL_switchon  " $HIFI_BB_ID 9 " " $HIFI_cmd 3429155674"
```

A3.5 HIFI_Total_power.tcl

```
tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID       1        " \
   " $HIFI_OBS_ID      1        " \
   referby HIFI_Set_OBS_ID_flg

   tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  1       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 1       RAW " \
   " $HIF_N_HRS_INTEGR 1       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
```

```
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  2048    RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  2048    RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  2048    RAW " \
   " $HIF_HRS_RSHIFT   0       RAW " \
   " $HIF_WBS_RSHIFT   0       RAW " \
   " $HIF_HRSH_SEL     255     RAW " \
   " $HIF_HRSV_SEL     255     RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354560     RAW "

waittime 15

tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID       1        " \
   " $HIFI_OBS_ID      2        " \
   referby HIFI_Set_OBS_ID_flg

tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  1       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 1       RAW " \
   " $HIF_N_HRS_INTEGR 1       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  2048    RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  2048    RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  2048    RAW " \
   " $HIF_HRS_RSHIFT   1       RAW " \
   " $HIF_WBS_RSHIFT   1       RAW " \
   " $HIF_HRSH_SEL     255     RAW " \
   " $HIF_HRSV_SEL     255     RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354561     RAW "

waittime 15

tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID       1        " \
   " $HIFI_OBS_ID      3        " \
   referby HIFI_Set_OBS_ID_flg

tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  1       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 1       RAW " \
```

```
   " $HIF_N_HRS_INTEGR 1       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  2048    RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  2048    RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  2048    RAW " \
   " $HIF_HRS_RSHIFT   2       RAW " \
   " $HIF_WBS_RSHIFT   2       RAW " \
   " $HIF_HRSH_SEL     255     RAW " \
   " $HIF_HRSV_SEL     255     RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354562     RAW "

waittime 15

tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID       1        " \
   " $HIFI_OBS_ID      4        " \
   referby HIFI_Set_OBS_ID_flg


tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  1       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 1       RAW " \
   " $HIF_N_HRS_INTEGR 1       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  2048    RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  2048    RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  2048    RAW " \
   " $HIF_HRS_RSHIFT   16      RAW " \
   " $HIF_WBS_RSHIFT   16      RAW " \
   " $HIF_HRSH_SEL     255     RAW " \
   " $HIF_HRSV_SEL     255     RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354563     RAW "

waittime 15

tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID       1        " \
   " $HIFI_OBS_ID      5        " \
   referby HIFI_Set_OBS_ID_flg
```

```
tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  2       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 2       RAW " \
   " $HIF_N_HRS_INTEGR 2       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  2048    RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  2048    RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  2048    RAW " \
   " $HIF_HRS_RSHIFT   0       RAW " \
   " $HIF_WBS_RSHIFT   0       RAW " \
   " $HIF_HRSH_SEL     255     RAW " \
   " $HIF_HRSV_SEL     255     RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354564     RAW "

waittime 15

tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID       1       " \
   " $HIFI_OBS_ID      6       " \
   referby HIFI_Set_OBS_ID_flg

tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  2       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 2       RAW " \
   " $HIF_N_HRS_INTEGR 2       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  2048    RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  2048    RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  2048    RAW " \
   " $HIF_HRS_RSHIFT   1       RAW " \
   " $HIF_WBS_RSHIFT   1       RAW " \
   " $HIF_HRSH_SEL     255     RAW " \
   " $HIF_HRSV_SEL     255     RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354565     RAW "

waittime 15
```

```
tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID      1        " \
   " $HIFI_OBS_ID     7        " \
   referby HIFI_Set_OBS_ID_flg

tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  4       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 4       RAW " \
   " $HIF_N_HRS_INTEGR 4       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  2048    RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  2048    RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  2048    RAW " \
   " $HIF_HRS_RSHIFT   0       RAW " \
   " $HIF_WBS_RSHIFT   0       RAW " \
   " $HIF_HRSH_SEL     255     RAW " \
   " $HIF_HRSV_SEL     255     RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354566     RAW "

waittime 25

tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID      1        " \
   " $HIFI_OBS_ID     8        " \
   referby HIFI_Set_OBS_ID_flg

tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  4       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 4       RAW " \
   " $HIF_N_HRS_INTEGR 4       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  2048    RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  2048    RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  2048    RAW " \
   " $HIF_HRS_RSHIFT   2       RAW " \
   " $HIF_WBS_RSHIFT   2       RAW " \
   " $HIF_HRSH_SEL     255     RAW " \
   " $HIF_HRSV_SEL     255     RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg
```

```
waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354567     RAW "

waittime 25

tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID       1        " \
   " $HIFI_OBS_ID      9        " \
   referby HIFI_Set_OBS_ID_flg

tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  4       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 4       RAW " \
   " $HIF_N_HRS_INTEGR 4       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  0       RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  0       RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  0       RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  0       RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  0       RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  0       RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  0       RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  0       RAW " \
   " $HIF_HRS_RSHIFT   2       RAW " \
   " $HIF_WBS_RSHIFT   2       RAW " \
   " $HIF_HRSH_SEL     0       RAW " \
   " $HIF_HRSV_SEL     0       RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354568     RAW "

waittime 25

tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID       1        " \
   " $HIFI_OBS_ID      10       " \
   referby HIFI_Set_OBS_ID_flg

tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  4       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 4       RAW " \
   " $HIF_N_HRS_INTEGR 4       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  0       RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  0       RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  0       RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  0       RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  0       RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  0       RAW " \
   " $HIF_HRS_RSHIFT   2       RAW " \
   " $HIF_WBS_RSHIFT   2       RAW " \
```

```
   " $HIF_HRSH_SEL     160     RAW " \
   " $HIF_HRSV_SEL     10      RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354569     RAW "

waittime 25

tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID       1       " \
   " $HIFI_OBS_ID      11      " \
   referby HIFI_Set_OBS_ID_flg

tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  4       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 4       RAW " \
   " $HIF_N_HRS_INTEGR 4       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 5120    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  100     RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  100     RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  100     RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  100     RAW " \
   " $HIF_HRS_RSHIFT   2       RAW " \
   " $HIF_WBS_RSHIFT   2       RAW " \
   " $HIF_HRSH_SEL     170     RAW " \
   " $HIF_HRSV_SEL     85      RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       2684354570     RAW "

waittime 25

tcsend $HIFI_Set_OBS_ID checks {NONE} ack {ACCEPT}\
   " $HIFI_BB_ID       1       " \
   " $HIFI_OBS_ID      12      " \
   referby HIFI_Set_OBS_ID_flg

tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       2       RAW " \
   " $HIF_N_WBS_START  4       RAW " \
   " $HIF_R_HRS        8       RAW " \
   " $HIF_N_WBS_INTEGR 4       RAW " \
   " $HIF_N_HRS_INTEGR 32      RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    1205    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  2048    RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
```

```
   " $HIF_WBSV_WIDTH3  2048    RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  2048    RAW " \
   " $HIF_HRS_RSHIFT   0       RAW " \
   " $HIF_WBS_RSHIFT   0       RAW " \
   " $HIF_HRSH_SEL     255     RAW " \
   " $HIF_HRSV_SEL     255     RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_total_power  ack {ACCEPT COMPLETE} " $HIFI_BB_ID       12303291       RAW "

waittime 15
```

A3.6 HIFI_Fast_Chop.tcl

```
tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  1       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 1       RAW " \
   " $HIF_N_HRS_INTEGR 1       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    2005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
   " $HIF_WBSV_OFFSET1 0       RAW " \
   " $HIF_WBSV_WIDTH1  2048    RAW " \
   " $HIF_WBSV_OFFSET2 2048    RAW " \
   " $HIF_WBSV_WIDTH2  2048    RAW " \
   " $HIF_WBSV_OFFSET3 4096    RAW " \
   " $HIF_WBSV_WIDTH3  2048    RAW " \
   " $HIF_WBSV_OFFSET4 6144    RAW " \
   " $HIF_WBSV_WIDTH4  2048    RAW " \
   " $HIF_HRS_RSHIFT   0       RAW " \
   " $HIF_WBS_RSHIFT   0       RAW " \
   " $HIF_HRSH_SEL     255     RAW " \
   " $HIF_HRSV_SEL     255     RAW " \
   " $HIF_WBS_packing  24_bits_format ENG " \
   referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_fast_chop  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       2952790017     RAW " \
   " $HIF_CPR_CH_ROT1  0       ENG " \
   " $HIF_CPR_CH_ROT2  0       ENG " \
   " $HIF_N_WBS1       2       RAW " \
   " $HIF_N_HRS_TRANS  4       RAW " \
   referby HIFI_Spectr_fast_chop_flg

waittime 40

tcsend $HIFI_config_spectroscopy  ack {ACCEPT COMPLETE}\
   " $HIFI_BB_ID       1       RAW " \
   " $HIF_N_WBS_START  4       RAW " \
   " $HIF_R_HRS        1       RAW " \
   " $HIF_N_WBS_INTEGR 4       RAW " \
   " $HIF_N_HRS_INTEGR 4       RAW " \
   " $HIF_DEL_HRS      5       RAW " \
   " $HIF_DEL_WBS      5       RAW " \
   " $HIF_T_ACC_WBS    2005    RAW " \
   " $HIF_T_ACC_HRS    100     RAW " \
   " $HIF_WBSH_OFFSET1 0       RAW " \
   " $HIF_WBSH_WIDTH1  2048    RAW " \
   " $HIF_WBSH_OFFSET2 2048    RAW " \
   " $HIF_WBSH_WIDTH2  2048    RAW " \
   " $HIF_WBSH_OFFSET3 4096    RAW " \
   " $HIF_WBSH_WIDTH3  2048    RAW " \
   " $HIF_WBSH_OFFSET4 6144    RAW " \
   " $HIF_WBSH_WIDTH4  2048    RAW " \
```

```
    " $HIF_WBSV_OFFSET1 0       RAW " \
    " $HIF_WBSV_WIDTH1  2048    RAW " \
    " $HIF_WBSV_OFFSET2 2048    RAW " \
    " $HIF_WBSV_WIDTH2  2048    RAW " \
    " $HIF_WBSV_OFFSET3 4096    RAW " \
    " $HIF_WBSV_WIDTH3  2048    RAW " \
    " $HIF_WBSV_OFFSET4 6144    RAW " \
    " $HIF_WBSV_WIDTH4  2048    RAW " \
    " $HIF_HRS_RSHIFT   2       RAW " \
    " $HIF_WBS_RSHIFT   2       RAW " \
    " $HIF_HRSH_SEL     255     RAW " \
    " $HIF_HRSV_SEL     255     RAW " \
    " $HIF_WBS_packing  24_bits_format ENG " \
    referby HIFI_config_spectroscopy_flg

waittime 3

tcsend $HIFI_Spectr_fast_chop  ack {ACCEPT COMPLETE}\
    " $HIFI_BB_ID       2952790018     RAW " \
    " $HIF_CPR_CH_ROT1  0       ENG " \
    " $HIF_CPR_CH_ROT2  0       ENG " \
    " $HIF_N_WBS1       4       RAW " \
    " $HIF_N_HRS_TRANS  4       RAW " \
    referby HIFI_Spectr_fast_chop_flg

waittime 40
```

A3.7 HIFI_disable_TM

```
tcsend $HIFI_enable_transmision  ack {ACCEPT}\
    " $HIF_N_TX_packets 5       RAW " \
    " $HIF_TX_packet_ID HRSH_tune       ENG " \
    " $HIF_TX_packet_ID HRSV_tune       ENG " \
    " $HIF_TX_packet_ID WBSH_tune       ENG " \
    " $HIF_TX_packet_ID WBSV_tune       ENG " \
    " $HIF_TX_packet_ID MXMG_tune       ENG " \
    referby HIFI_enable_transmision_flg

waittime 1

tcsend $HIFI_report_transmision  ack {ACCEPT}\
    referby HIFI_report_TM_flg

waittime 1

tcsend $HIFI_disable_transmision  ack {ACCEPT}\
    " $HIF_N_TX_packets 20      RAW " \
    " $HIF_TX_packet_ID HRSH1_start     ENG " \
    " $HIF_TX_packet_ID HRSH1_SD24      ENG " \
    " $HIF_TX_packet_ID HRSH2_start     ENG " \
    " $HIF_TX_packet_ID HRSH2_SD24      ENG " \
    " $HIF_TX_packet_ID HRSV1_start     ENG " \
    " $HIF_TX_packet_ID HRSV1_SD24      ENG " \
    " $HIF_TX_packet_ID HRSV2_start     ENG " \
    " $HIF_TX_packet_ID HRSV2_SD24      ENG " \
    " $HIF_TX_packet_ID WBSH1_start     ENG " \
    " $HIF_TX_packet_ID WBSH1_SD16      ENG " \
    " $HIF_TX_packet_ID WBSH1_SD24      ENG " \
    " $HIF_TX_packet_ID WBSH2_start     ENG " \
    " $HIF_TX_packet_ID WBSH2_SD16      ENG " \
    " $HIF_TX_packet_ID WBSH2_SD24      ENG " \
    " $HIF_TX_packet_ID WBSV1_start     ENG " \
    " $HIF_TX_packet_ID WBSV1_SD16      ENG " \
    " $HIF_TX_packet_ID WBSV1_SD24      ENG " \
    " $HIF_TX_packet_ID WBSV2_start     ENG " \
    " $HIF_TX_packet_ID WBSV2_SD16      ENG " \
    " $HIF_TX_packet_ID WBSV2_SD24      ENG " \
    referby HIFI_disable_TM_flg

waittime 1

tcsend $HIFI_report_transmision  ack {ACCEPT}\
    referby HIFI_report_TM_flg


tcsend $HIFI_disable_transmision  ack {ACCEPT}\
    " $HIF_N_TX_packets 58      RAW " \
    " $HIF_TX_packet_ID HIFI_HK_REV4    ENG " \
```

```
   " $HIF_TX_packet_ID HIFI_FCUrev1    ENG " \
   " $HIF_TX_packet_ID Essential_HK    ENG " \
   " $HIF_TX_packet_ID HRSH1_IF_pwr    ENG " \
   " $HIF_TX_packet_ID HRSH2_IF_pwr    ENG " \
   " $HIF_TX_packet_ID HRSV1_IF_pwr    ENG " \
   " $HIF_TX_packet_ID HRSV2_IF_pwr    ENG " \
   " $HIF_TX_packet_ID WBSH1_IF_pwr    ENG " \
   " $HIF_TX_packet_ID WBSH2_IF_pwr    ENG " \
   " $HIF_TX_packet_ID WBSV1_IF_pwr    ENG " \
   " $HIF_TX_packet_ID WBSV2_IF_pwr    ENG " \
   " $HIF_TX_packet_ID HIFI_READY      ENG " \
   " $HIF_TX_packet_ID PM_failed       ENG " \
   " $HIF_TX_packet_ID DM_failed       ENG " \
   " $HIF_TX_packet_ID EEPROM_fail     ENG " \
   " $HIF_TX_packet_ID HIFI_TC_ver     ENG " \
   " $HIF_TX_packet_ID HIFI_load_ee    ENG " \
   " $HIF_TX_packet_ID HIFI_load_dm    ENG " \
   " $HIF_TX_packet_ID HIFI_boot_dm    ENG " \
   " $HIF_TX_packet_ID runtime_err     ENG " \
   " $HIF_TX_packet_ID DHTR_H_OOL      ENG " \
   " $HIF_TX_packet_ID DHTR_V_OOL      ENG " \
   " $HIF_TX_packet_ID LASER_H_OOL     ENG " \
   " $HIF_TX_packet_ID LASER_V_OOL     ENG " \
   " $HIF_TX_packet_ID MX_H_nonresp    ENG " \
   " $HIF_TX_packet_ID MX_V_nonresp    ENG " \
   " $HIF_TX_packet_ID Chop_nonresp    ENG " \
   " $HIF_TX_packet_ID HIFI_peakup     ENG " \
   " $HIF_TX_packet_ID FCU_report3     ENG " \
   " $HIF_TX_packet_ID HRH_att_lo3     ENG " \
   " $HIF_TX_packet_ID HRH_blocks3     ENG " \
   " $HIF_TX_packet_ID HRV_att_lo3     ENG " \
   " $HIF_TX_packet_ID HRV_blocks3     ENG " \
   " $HIF_TX_packet_ID WBS_H_rep3      ENG " \
   " $HIF_TX_packet_ID WBS_V_rep3      ENG " \
   " $HIF_TX_packet_ID LCU_ch1a_rep    ENG " \
   " $HIF_TX_packet_ID LCU_ch2a_rep    ENG " \
   " $HIF_TX_packet_ID LCU_ch3a_rep    ENG " \
   " $HIF_TX_packet_ID LCU_ch3b_rep    ENG " \
   " $HIF_TX_packet_ID LCU_ch4a_rep    ENG " \
   " $HIF_TX_packet_ID LCU_ch5a_rep    ENG " \
   " $HIF_TX_packet_ID LCU_ch6a_rep    ENG " \
   " $HIF_TX_packet_ID FCU_pwr_rep3    ENG " \
   " $HIF_TX_packet_ID Spectr_rep      ENG " \
   " $HIF_TX_packet_ID Spectr_slow     ENG " \
   " $HIF_TX_packet_ID mem_dump        ENG " \
   " $HIF_TX_packet_ID mem_check       ENG " \
   " $HIF_TX_packet_ID Time_verif      ENG " \
   " $HIF_TX_packet_ID TM_gen_stat     ENG " \
   " $HIF_TX_packet_ID HIFI_Conn       ENG " \
   " $HIF_TX_packet_ID HRSH1_start     ENG " \
   " $HIF_TX_packet_ID HRSH1_SD24      ENG " \
   " $HIF_TX_packet_ID HRSH2_start     ENG " \
   " $HIF_TX_packet_ID HRSH2_SD24      ENG " \
   " $HIF_TX_packet_ID HRSV1_start     ENG " \
   " $HIF_TX_packet_ID HRSV1_SD24      ENG " \
   " $HIF_TX_packet_ID HRSV2_start     ENG " \
   " $HIF_TX_packet_ID HRSV2_SD24      ENG " \
   referby HIFI_disable_TM_flg

waittime 1

tcsend $HIFI_report_transmision  ack {ACCEPT}\
   referby HIFI_report_TM_flg
```

A3.8 HIFI_Enable_TM

```
tcsend $HIFI_enable_transmision  ack {ACCEPT}\
   " $HIF_N_TX_packets 1       RAW " \
   " $HIF_TX_packet_ID runtime_err     ENG " \
   referby HIFI_enable_TM_flg

waittime 1

tcsend $HIFI_report_transmision  ack {ACCEPT}\
   referby HIFI_report_TM_flg

waittime 1

tcsend $HIFI_enable_transmision  ack {ACCEPT}\
   " $HIF_N_TX_packets 75      RAW " \
```

```
   " $HIF_TX_packet_ID HIFI_HK_REV4   ENG " \
   " $HIF_TX_packet_ID HIFI_FCUrev1   ENG " \
   " $HIF_TX_packet_ID Essential_HK   ENG " \
   " $HIF_TX_packet_ID HRSH1_IF_pwr   ENG " \
   " $HIF_TX_packet_ID HRSH2_IF_pwr   ENG " \
   " $HIF_TX_packet_ID HRSV1_IF_pwr   ENG " \
   " $HIF_TX_packet_ID HRSV2_IF_pwr   ENG " \
   " $HIF_TX_packet_ID WBSH1_IF_pwr   ENG " \
   " $HIF_TX_packet_ID WBSH2_IF_pwr   ENG " \
   " $HIF_TX_packet_ID WBSV1_IF_pwr   ENG " \
   " $HIF_TX_packet_ID WBSV2_IF_pwr   ENG " \
   " $HIF_TX_packet_ID HIFI_READY     ENG " \
   " $HIF_TX_packet_ID PM_failed      ENG " \
   " $HIF_TX_packet_ID DM_failed      ENG " \
   " $HIF_TX_packet_ID EEPROM_fail    ENG " \
   " $HIF_TX_packet_ID HIFI_TC_ver    ENG " \
   " $HIF_TX_packet_ID HIFI_load_ee   ENG " \
   " $HIF_TX_packet_ID HIFI_load_dm   ENG " \
   " $HIF_TX_packet_ID HIFI_boot_dm   ENG " \
   " $HIF_TX_packet_ID runtime_err    ENG " \
   " $HIF_TX_packet_ID DHTR_H_OOL     ENG " \
   " $HIF_TX_packet_ID DHTR_V_OOL     ENG " \
   " $HIF_TX_packet_ID LASER_H_OOL    ENG " \
   " $HIF_TX_packet_ID LASER_V_OOL    ENG " \
   " $HIF_TX_packet_ID MX_H_nonresp   ENG " \
   " $HIF_TX_packet_ID MX_V_nonresp   ENG " \
   " $HIF_TX_packet_ID Chop_nonresp   ENG " \
   " $HIF_TX_packet_ID HIFI_peakup    ENG " \
   " $HIF_TX_packet_ID FCU_report3    ENG " \
   " $HIF_TX_packet_ID HRH_att_lo3    ENG " \
   " $HIF_TX_packet_ID HRH_blocks3    ENG " \
   " $HIF_TX_packet_ID HRV_att_lo3    ENG " \
   " $HIF_TX_packet_ID HRV_blocks3    ENG " \
   " $HIF_TX_packet_ID WBS_H_rep3     ENG " \
   " $HIF_TX_packet_ID WBS_V_rep3     ENG " \
   " $HIF_TX_packet_ID LCU_ch1a_rep   ENG " \
   " $HIF_TX_packet_ID LCU_ch2a_rep   ENG " \
   " $HIF_TX_packet_ID LCU_ch3a_rep   ENG " \
   " $HIF_TX_packet_ID LCU_ch3b_rep   ENG " \
   " $HIF_TX_packet_ID LCU_ch4a_rep   ENG " \
   " $HIF_TX_packet_ID LCU_ch5a_rep   ENG " \
   " $HIF_TX_packet_ID LCU_ch6a_rep   ENG " \
   " $HIF_TX_packet_ID FCU_pwr_rep3   ENG " \
   " $HIF_TX_packet_ID Spectr_rep     ENG " \
   " $HIF_TX_packet_ID Spectr_slow    ENG " \
   " $HIF_TX_packet_ID mem_dump       ENG " \
   " $HIF_TX_packet_ID mem_check      ENG " \
   " $HIF_TX_packet_ID Time_verif     ENG " \
   " $HIF_TX_packet_ID TM_gen_stat    ENG " \
   " $HIF_TX_packet_ID HIFI_Conn      ENG " \
   " $HIF_TX_packet_ID HRSH1_start    ENG " \
   " $HIF_TX_packet_ID HRSH1_SD24     ENG " \
   " $HIF_TX_packet_ID HRSH2_start    ENG " \
   " $HIF_TX_packet_ID HRSH2_SD24     ENG " \
   " $HIF_TX_packet_ID HRSV1_start    ENG " \
   " $HIF_TX_packet_ID HRSV1_SD24     ENG " \
   " $HIF_TX_packet_ID HRSV2_start    ENG " \
   " $HIF_TX_packet_ID HRSV2_SD24     ENG " \
   " $HIF_TX_packet_ID WBSH1_SD16     ENG " \
   " $HIF_TX_packet_ID WBSH1_SD24     ENG " \
   " $HIF_TX_packet_ID WBSH2_start    ENG " \
   " $HIF_TX_packet_ID WBSH2_SD16     ENG " \
   " $HIF_TX_packet_ID WBSH2_SD24     ENG " \
   " $HIF_TX_packet_ID WBSV1_start    ENG " \
   " $HIF_TX_packet_ID WBSV1_SD16     ENG " \
   " $HIF_TX_packet_ID WBSV1_SD24     ENG " \
   " $HIF_TX_packet_ID WBSV2_start    ENG " \
   " $HIF_TX_packet_ID WBSV2_SD16     ENG " \
   " $HIF_TX_packet_ID WBSV2_SD24     ENG " \
   " $HIF_TX_packet_ID HRSH_tune      ENG " \
   " $HIF_TX_packet_ID HRSV_tune      ENG " \
   " $HIF_TX_packet_ID WBSH_tune      ENG " \
   " $HIF_TX_packet_ID WBSV_tune      ENG " \
   " $HIF_TX_packet_ID MXMG_tune      ENG " \
   referby HIFI_enable_TM_flg

waittime 1

tcsend $HIFI_report_transmision  ack {ACCEPT}\
   referby HIFI_report_TM_flg

waittime 1
```

```
tcsend $HIFI_enable_transmision  ack {ACCEPT}\
   " $HIF_N_TX_packets 8       RAW " \
   " $HIF_TX_packet_ID alarms  ENG " \
   " $HIF_TX_packet_ID memdump ENG " \
   " $HIF_TX_packet_ID memcheck       ENG " \
   " $HIF_TX_packet_ID time_verif     ENG " \
   " $HIF_TX_packet_ID Transmission   ENG " \
   " $HIF_TX_packet_ID Connection     ENG " \
   " $HIF_TX_packet_ID Science ENG " \
   referby HIFI_enable_TM_flg

waittime 1

tcsend $HIFI_report_transmision  ack {ACCEPT}\
   referby HIFI_report_TM_flg

waittime 1
```