

Load SPIRE OBS from ASW
File: H_FCP_OBS_5110.xls
Author: Liviu Stefanov



Procedure Summary

Objectives

This Herschel OBSM nominal procedure is used to execute the SPIRE OBS full image upload from the Application Software (ASW). It is called by the FOP SPIRE procedures H_FCP_SPI_CLOM and H_FCP_SPI_CLOA.

The OBS image is loaded into the SPIRE DPU PM-High memory and the image integrity after upload is checked via checksum calculation and verification.

The copying of the OBS image from PM-High to PM-Low and OBS restart is executed in the calling procedure H_FCP_SPI_CLOM or H_FCP_SPI_CLOA.

The calling procedure also includes the PM-Low OBS image checksum verification and updated OBS version numbers verification.

This procedure assumes that the memory load and memory check command stacks have already been generated using the OBSM system and are ready for loading on the Manual Stack. The command stack generation activity is not covered by this procedure.

Note: Patching (as alternative to full image upload) of the SPIRE OBS from the ASW can be conducted via procedure H_FCP_OBS_5111.

Summary of Constraints

CDMU in Operational Mode

- SPIRE DPU is ON
- SPIRE ASW running

Memory areas are Loaded through TC(6,2) and Checked through TC(6,9); this TCs will be delayed when there is an ongoing:

- TC(6,2) Load Memory Using Absolute Addresses
- TC(6,5) Dump Memory Using Absolute Addresses
- TC(6,9) Check Memory Using Absolute Addresses
- TC(8,4,1,1) Copy Memory

Spacecraft Configuration

Start of Procedure

CDMU in Operational Mode

- SPIRE DPU is ON
- SPIRE ASW running

End of Procedure

Same as start except:

- New SPIRE OBS image loaded in DPU PM-High memory

Reference File(s)

Input Command Sequences

Output Command Sequences

OFCP5110

Load SPIRE OBS from ASW
 File: H_FCP_OBS_5110.xls
 Author: Liviu Stefanov



Referenced Displays

ANDs GRDs SLDs
 SA_1_559
 SAM4_500

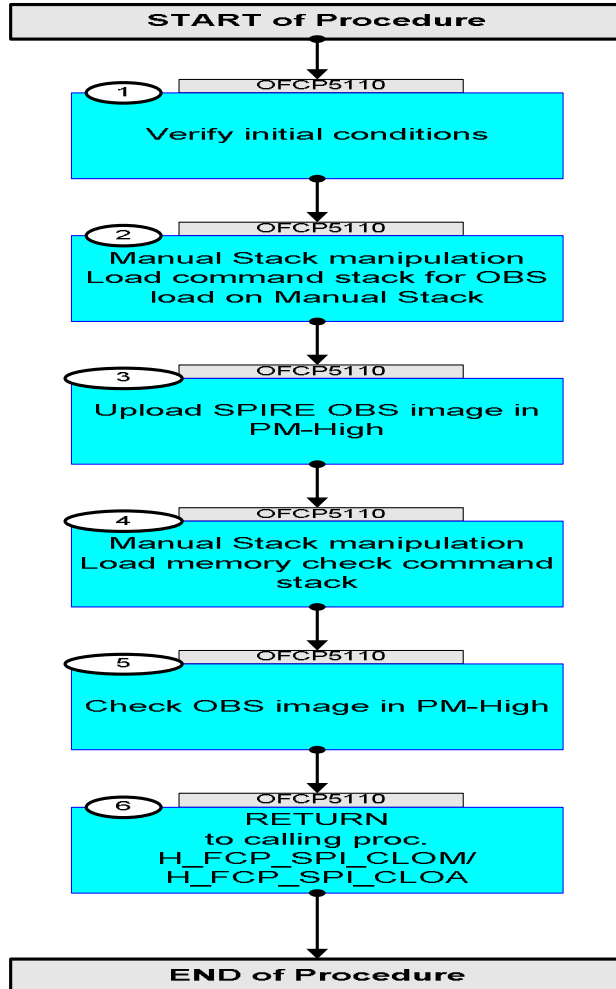
Configuration Control Information

DATE	FOP ISSUE	VERSION	MODIFICATION DESCRIPTION	AUTHOR	SPR REF
30/01/08	1	1	Created	Istefanov-hp	
			1. step 2 and sub-steps updated to separate patch stack load for Prime and Redundant 2. step 2 updated for SPIRE OBS v.3.0.B 3. added current step 4 to separate check stack load from patch stack load 4. added current step 6 to include return to calling procedure	Istefanov-hp	
07/10/09		2		Istefanov-hp	
07/10/09		3	1. corrected some Cell Format Pattern problems	Istefanov-hp	
07/10/09	2.5	4	1. corrected some Cell Format Pattern problems	Istefanov-hp	
11/11/09	3	5	Updated in line with ASW v4.0. Command stack names, and checks updated.	m.baker-hp	

Load SPIRE OBS from ASW
File: H_FCP_OBS_5110.xls
Author: m.baker-hp



Procedure Flowchart Overview



Load SPIRE OBS from ASW File: H_FCP_OBS_5110.xls Author: m.baker-hp	 
---	--

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment
Beginning of Procedure					
	OFCP5110	TC Seq. Name : OFCP5110 () Load SPIRE OBS from ASW TimeTag Type: B Sub Schedule ID: <input type="checkbox"/>			
1		Verify initial conditions		Next Step: 2	
		Check: - SPIRE DPU in REDY mode - SPIRE ASW running			
		Instrument SOE to confirm SPIRE instrument mode			
		Note: Initial conditions are verified in calling procedure H_FCP_SPI_CLOM or H_FCP_SPI_CLOA.			
2		Manual Stack manipulation Load command stack for OBS load on Manual Stack		Next Step: 3	
		NOTE: The current procedure assumes that the memory load is performed using commands with immediate execution.			
		Select the File -> LoadStack option from the main menu of the Manual Stack window			
2.1		IF SPIRE Nominal			
		Select file SPDPRMPG_PI_XXXXYYY_N_NoModel_NoModel_YYYY_DDDThhmss.machine from directory /home/hmcsops/HPMCS/SESSION/current/data/CMD/STACKS/OBSM/SPDPRMPG as indicated by the OBSM engineer			
		IMPORTANT: XXXXYYY = Image ID(X) and Version(Y) - depend on image used for stack generation YYYY_DDD hhmss - depend on stack generation time machine - depends on the name of the machine used for stack generation			

Load SPIRE OBS from ASW File: H_FCP_OBS_5110.xls Author: m.baker-hp	 
---	--

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment
		File name examples - No model associated to the memory image: SPDPRMPR_PI_0002001_N_NoModel_NoModel_2007_254T123300.sun043 - CT SPDPRMPR1, ID 0003, Version 001 associated to the memory image: SPDPRMPR_PI_0002001_C_SPDPRMPR1_0003001_2007_337T093320.sun043			
		File name for version 4.0 - No model associated to the memory image: SPDPRMPG_PI_0003001_N_NoModel_NoModel_2009_308T155501.ws044			
2.2		ELSE SPIRE Redundant			
		Select file SPDPRMPR_PI_XXXXYYY_N_NoModel_NoModel_YYYY_DDDThhmmss.machine from directory /home/hmcsops/HPMCS/SESSION/current/data/CMD/STACKS/OBSM/SPDPRMPR as indicated by the OBSM engineer			
		IMPORTANT: XXXXYYY = Image ID(X) and Version(Y) - depend on image used for stack generation YYYY_DDD hhmmss - depend on stack generation time machine - depends on the name of the machine used for stack generation			
		File name examples - No model associated to the memory image: SPDPRMPR_PI_0002001_N_NoModel_NoModel_2007_254T123300.sun043 - CT SPDPRMPR1, ID 0003, Version 001 associated to the memory image: SPDPRMPR_PI_0002001_C_SPDPRMPR1_0003001_2007_337T093320.sun043			
2.3		Check memory load command stack loaded			

Load SPIRE OBS from ASW File: H_FCP_OBS_5110.xls Author: m.baker-hp	 
---	--

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment															
		For SPIRE OBS v.3.0.B: The start address of the SPDRMPG memory image used for memory load command stack generation is 00.0000 hex , and the last address in the image is 01.6E12 hex . The offset applied to the memory image for OBS upload in PM-High is 04.0000 hex . Consequently, the first address to be loaded is 04.0000 hex , and the last address is 05.6E12 hex .																		
		For SPIRE OBS v.4.0: The start address of the SPDRMPG memory image used for memory load command stack generation is 00.0000 hex , and the last address in the image is 1.6057 hex . The offset applied to the memory image for OBS upload in PM-High is 02.0000 hex . Consequently, the first address to be loaded is 02.0000 hex , and the last address is 03.6057 hex .																		
2.3.1		Check number of memory load commands in the stack																		
		Check that loaded stack contains 2374 TCs XC002998																		
2.3.2		Check Memory ID																		
		Display the Manual Stack in 'Full mode' and check that the Memory ID parameter in the XC002998 commands is set to 00 hex : Memory ID = 00 hex Note: The Memory ID of the target memory device is stored in the MSB of the 16-bit long Mem ID TC parameter. The LSB of the same parameter carries the most significant 8 bits of the Start Address.																		
		Execute Telecommand <div style="text-align: center;">SPIRE Memory Load</div> <i>Command Parameter(s) :</i> <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="padding-right: 20px;">Memory ID</td> <td style="padding-right: 20px;">XH000998</td> <td style="padding-right: 20px;">00xx hex</td> </tr> <tr> <td>Start Address</td> <td>XH001998</td> <td><hex> (Def)</td> </tr> <tr> <td>Length of Block</td> <td>XH003998</td> <td><dec> (Def)</td> </tr> <tr> <td>Var length octet string</td> <td>XH004998</td> <td><hex> (Def)</td> </tr> <tr> <td>Checksum</td> <td>XH005998</td> <td><hex> (Def)</td> </tr> </table> <i>TC Control Flags :</i> <div style="text-align: center;"> GBM IL DSE --Y -- --- </div> <i>Subsch. ID : 30</i> <i>Det. descr. : Load SPIRE Memory Using Absolute Addresses</i> This Telecommand will not be included in the export	Memory ID	XH000998	00xx hex	Start Address	XH001998	<hex> (Def)	Length of Block	XH003998	<dec> (Def)	Var length octet string	XH004998	<hex> (Def)	Checksum	XH005998	<hex> (Def)	XC002998	TC	
Memory ID	XH000998	00xx hex																		
Start Address	XH001998	<hex> (Def)																		
Length of Block	XH003998	<dec> (Def)																		
Var length octet string	XH004998	<hex> (Def)																		
Checksum	XH005998	<hex> (Def)																		

Load SPIRE OBS from ASW File: H_FCP_OBS_5110.xls Author: m.baker-hp	
---	--

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment															
2.3.3		Check start address and length of first command in the stack																		
		With the Manual Stack in 'Full mode', check the Start Address in the first XC002998 command: Start Address = 20000 hex Length = 38 dec Note: The Memory ID of the target memory device is stored in the MSB of the 16-bit long Mem ID TC parameter. The LSB of the same parameter carries the most significant 8 bits of the Start Address.																		
		Execute Telecommand <div style="text-align: right; margin-left: 200px;">SPIRE Memory Load</div> <div style="text-align: right; margin-left: 200px;">XC002998</div> <i>Command Parameter(s) :</i> <table style="margin-left: 100px; border: none;"> <tr><td style="padding-right: 20px;">Memory ID</td><td style="padding-right: 20px;">XH000998</td><td>0002 <hex></td></tr> <tr><td>Start Address</td><td>XH001998</td><td>0000 <hex></td></tr> <tr><td>Length of Block</td><td>XH003998</td><td>38 <dec></td></tr> <tr><td>Var length octet string</td><td>XH004998</td><td><hex> (Def)</td></tr> <tr><td>Checksum</td><td>XH005998</td><td><hex> (Def)</td></tr> </table> <i>TC Control Flags :</i> <div style="margin-left: 100px;">GBM IL DSE</div> <div style="margin-left: 100px;">--Y -- ---</div> <i>Subsch. ID :</i> 30 <i>Det. descr. :</i> Load SPIRE Memory Using Absolute Addresses	Memory ID	XH000998	0002 <hex>	Start Address	XH001998	0000 <hex>	Length of Block	XH003998	38 <dec>	Var length octet string	XH004998	<hex> (Def)	Checksum	XH005998	<hex> (Def)		TC	
Memory ID	XH000998	0002 <hex>																		
Start Address	XH001998	0000 <hex>																		
Length of Block	XH003998	38 <dec>																		
Var length octet string	XH004998	<hex> (Def)																		
Checksum	XH005998	<hex> (Def)																		
		This Telecommand will not be included in the export																		
2.3.4		Check start address and length of last command in the stack																		
		With the Manual Stack in 'Full mode', check the Start Address in the last XC002998 command: Start Address = 03.603E hex Length = 26 dec Note: The Memory ID of the target memory device is stored in the MSB of the 16-bit long Mem ID TC parameter. The LSB of the same parameter carries the most significant 8 bits of the Start Address.																		
		Execute Telecommand <div style="text-align: right; margin-left: 200px;">SPIRE Memory Load</div> <div style="text-align: right; margin-left: 200px;">XC002998</div> <i>Command Parameter(s) :</i> <table style="margin-left: 100px; border: none;"> <tr><td style="padding-right: 20px;">Memory ID</td><td style="padding-right: 20px;">XH000998</td><td>0003 <hex></td></tr> <tr><td>Start Address</td><td>XH001998</td><td>603E <hex></td></tr> <tr><td>Length of Block</td><td>XH003998</td><td>26 <dec></td></tr> <tr><td>Var length octet string</td><td>XH004998</td><td><hex> (Def)</td></tr> <tr><td>Checksum</td><td>XH005998</td><td><hex> (Def)</td></tr> </table> <i>TC Control Flags :</i> <div style="margin-left: 100px;">GBM IL DSE</div> <div style="margin-left: 100px;">--Y -- ---</div> <i>Subsch. ID :</i> 30 <i>Det. descr. :</i> Load SPIRE Memory Using Absolute Addresses	Memory ID	XH000998	0003 <hex>	Start Address	XH001998	603E <hex>	Length of Block	XH003998	26 <dec>	Var length octet string	XH004998	<hex> (Def)	Checksum	XH005998	<hex> (Def)		TC	
Memory ID	XH000998	0003 <hex>																		
Start Address	XH001998	603E <hex>																		
Length of Block	XH003998	26 <dec>																		
Var length octet string	XH004998	<hex> (Def)																		
Checksum	XH005998	<hex> (Def)																		
		This Telecommand will not be included in the export																		

Load SPIRE OBS from ASW File: H_FCP_OBS_5110.xls Author: m.baker-hp	 
---	--

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment
3		Upload SPIRE OBS image in PM-High		Next Step: 4	
		Uplink the XC002998 memory load commands with ARM-GO			
		For each TC XC002998 successfully executed on-board, the DPU HK counter TCEXEC should be incremented by one . After all XC002998 TCs have been sent, the value of the counter should be: incremented by 2374 for OBS v.4.0			
		Verify Telemetry TCEXEC SM03N500	= incremented by 2374 dec	AND=SA_1_559	
		For each TC XC002998 successfully executed on-board, a TM(1,1) and TM(1,7) packet shall be received on ground.			
3.1		IF SPIRE Prime			
		Verify Packet Reception P_TC_Acceptance_Report Packet Mnemonic : SP11TCAR0500 APID : 1280 Type : 1 Subtype : 1 PI1 : PI2 :			
		Verify Packet Reception P_TC_Execution_Completed_Report Packet Mnemonic : SP15TCECR500 APID : 1280 Type : 1 Subtype : 7 PI1 : PI2 :			
3.2		IF SPIRE Redundant			
		Verify Packet Reception R_TC_Acceptance_Report Packet Mnemonic : SP11TCAR0500 APID : 1281 Type : 1 Subtype : 1 PI1 : PI2 :			
		Verify Packet Reception R_TC_Execution_Completed_Report Packet Mnemonic : SP15TCECR500 APID : 1281 Type : 1 Subtype : 7 PI1 : PI2 :			

Load SPIRE OBS from ASW File: H_FCP_OBS_5110.xls Author: m.baker-hp	 
---	--

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment
4		Manual Stack manipulation Load memory check command stack		Next Step: 5	
		Select the File -> LoadStack option from the main menu of the Manual Stack window			
4.1		IF SPIRE Nominal			
		Select file SPDPRMPG_CI_XXXXYYY_N_NoModel_NoModel_YYYY_DDDThhmmss.machine from directory /home/hmcsofs/HPMCS/SESSION/current/data/CMD/STACKS/OBSM/SPDPRMPG as indicated by the OBSM engineer			
		IMPORTANT: XXXXYYY = Image ID(X) and Version(Y) - depend on image used for stack generation YYYY_DDD hhmmss - depend on stack generation time machine - depends on the name of the machine used for stack generation			
		File name examples - No model associated to the memory image: SPDPRMPG_CI_0002001_N_NoModel_NoModel_2007_254T123300.sun043 - CT SPDPRMPG1, ID 0003, Version 001 associated to the memory image: SPDPRMPG_CI_0002001_C_SPDPRMPG1_0003001_2007_337T093320.sun043			
		File name for version 4.0 - No model associated to the memory image: SPDPRMPG_CI_0003001_N_NoModel_NoModel_2009_309T125059.ws044			
4.2		ELSE SPIRE Redundant			

Load SPIRE OBS from ASW File: H_FCP_OBS_5110.xls Author: m.baker-hp	 
---	--

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment
		Select file SPDPRMPR_CI_XXXXYYY_N_NoModel_NoModel_YYYY_DDDThhmmss.machine from directory /home/hmcsops/HPMCS/SESSION/current/data/CMD/STACKS/OBSM/SPDPRMPR as indicated by the OBSM engineer			
		IMPORTANT: XXXXYYY = Image ID(X) and Version(Y) - depend on image used for stack generation YYYY_DDD hhmmss - depend on stack generation time machine - depends on the name of the machine used for stack generation			
		File name examples - No model associated to the memory image: SPDPRMPR_CI_0002001_N_NoModel_NoModel_2007_254T123300.sun043 - CT SPDPRMPR1, ID 0003, Version 001 associated to the memory image: SPDPRMPR_CI_0002001_C_SPDPRMPR1_0003001_2007_337T093320.sun043			
4.3		Check memory check command stack loaded			
		Check that 3 SCM02500 memory check commands have been loaded on the Manual Stack.			
		Display the Manual Stack in 'Full mode' and check the following addresses and lengths are covered by the SCM02500 commands:			
		For SPIRE OBS v.4.0 : Memory ID = 00 hex Start Address = 02.0000 hex End Address = 03.6057 hex			
		Note: The ' length ' parameter of the memory check command is a 16-bit long parameter. A memory check TC can cover a number of 65535 dec (FFFF hex) SAUs.			
		Execute Telecommand <div style="text-align: right; margin-right: 20px;"> CHECK_MEMORY </div> <div style="text-align: right; margin-right: 20px;"> SCM02500 </div> Command Parameter(s) : <div style="display: flex; justify-content: space-between; margin-left: 40px;"> <div> MEMORYID_CHECKMEM STARTADDR_CHECKMEM NSAU_CHECKMEM </div> <div> SPM9N500 SPMAN500 SPMBN500 </div> <div> 0002 <hex> 4000 <hex> 174B <hex> </div> </div> TC Control Flags : <div style="display: flex; justify-content: space-between; margin-left: 40px;"> <div> GBM IL DSE --Y -- --- </div> </div> Subsch. ID : 370 Det. descr. : CHECK MEMORY USING ABSOLUTE ADDRESSES This Telecommand will not be included in the export		TC	

Load SPIRE OBS from ASW
 File: H_FCP_OBS_5110.xls
 Author: m.baker-hp



Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment
		Execute Telecommand <p style="text-align: right;">CHECK_MEMORY</p> Command Parameter(s) : MEMORYID_CHECKMEM SPM9N500 STARTADDR_CHECKMEM SPMAN500 NSAU_CHECKMEM SPMBN500 TC Control Flags : <p style="text-align: right;">GBM IL DSE --Y -- ---</p> Subsch. ID : 370 Det. descr. : CHECK MEMORY USING ABSOLUTE ADDRESSES This Telecommand will not be included in the export	SCM02500	TC	
		Execute Telecommand <p style="text-align: right;">CHECK_MEMORY</p> Command Parameter(s) : MEMORYID_CHECKMEM SPM9N500 STARTADDR_CHECKMEM SPMAN500 NSAU_CHECKMEM SPMBN500 TC Control Flags : <p style="text-align: right;">GBM IL DSE --Y -- ---</p> Subsch. ID : 370 Det. descr. : CHECK MEMORY USING ABSOLUTE ADDRESSES This Telecommand will not be included in the export	SCM02500	TC	
5		Check OBS image in PM-High		Next Step: 6	
		For each TC(6,9), a TM(6,10) packet shall be received on ground.			
5.1		Command and verify the first checksum			
		Uplink the first SCM02500 memory check commands with ARM-GO			
5.1.1		Verify reception and contents of TM(6,10)			
		Note: A TM(6,10) packet will be received for each memory check command uplinked.			
5.1.1.1		IF SPIRE Prime			
		Verify Packet Reception Memory_Check_Absolute_Addresses Packet Mnemonic : SMEMCHK00500 APID : 1280 Type : 6 Subtype : 10 PI1 : PI2 :			
		Verify Telemetry <p style="text-align: right;">MEMORYID_MCHK SMM8N500</p>	= 0002 <hex>	AND=SAM4_500	

Load SPIRE OBS from ASW File: H_FCP_OBS_5110.xls Author: m.baker-hp	
---	--

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment
		Verify Telemetry STARTADDR_MCHK SMM9N500	= 4000 <hex>	AND=SAM4_500	
		Verify Telemetry NSAU_MCHK SMMAN500	= 174B <hex>	AND=SAM4_500	
		Verify Telemetry CHK_MCHK SMMBN500		AND=SAM4_500	
5.1.1.2		ELSE SPIRE Redundant			
		Verify Packet Reception R_Memory_Check_Absolute_Addresses Packet Mnemonic : SMEMCHK00500 APID : 1281 Type : 6 Subtype : 10 PI1 : PI2 :			
		Verify Telemetry MEMORYID_MCHK SMM8N500	= 0002 <hex>	AND=SAM4_500	
		Verify Telemetry STARTADDR_MCHK SMM9N500	= 4000 <hex>	AND=SAM4_500	
		Verify Telemetry NSAU_MCHK SMMAN500	= 174B <hex>	AND=SAM4_500	
		Verify Telemetry CHK_MCHK SMMBN500		AND=SAM4_500	
5.1.2		Verify checksum value			
		Check the received checksum against the expected value			
		Verify Telemetry CHK_MCHK SMMBN500	= expected value	AND=SAM4_500	
5.2		Command and verify the second checksum			
		Uplink the second SCM02500 memory check commands with ARM-GO			
5.2.1		Verify reception and contents of TM(6,10)			
		Note: A TM(6,10) packet will be received for each memory check command uplinked.			

Load SPIRE OBS from ASW
 File: H_FCP_OBS_5110.xls
 Author: m.baker-hp




Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment
5.2.1.1		IF SPIRE Prime			
		Verify Packet Reception Memory_Check_Absolute_Addresses Packet Mnemonic : SMEMCHK00500 APID : 1280 Type : 6 Subtype : 10 PI1 : PI2 :			
		Verify Telemetry MEMORYID_MCHK SMM8N500	= 0002 <hex>	AND=SAM4_500	
		Verify Telemetry STARTADDR_MCHK SMM9N500	= 6000 <hex>	AND=SAM4_500	
		Verify Telemetry NSAU_MCHK SMMAN500	= FFFF <hex>	AND=SAM4_500	
		Verify Telemetry CHK_MCHK SMMBN500		AND=SAM4_500	
5.2.1.2		ELSE SPIRE Redundant			
		Verify Packet Reception R_Memory_Check_Absolute_Addresses Packet Mnemonic : SMEMCHK00500 APID : 1281 Type : 6 Subtype : 10 PI1 : PI2 :			
		Verify Telemetry MEMORYID_MCHK SMM8N500	= 0002 <hex>	AND=SAM4_500	
		Verify Telemetry STARTADDR_MCHK SMM9N500	= 6000 <hex>	AND=SAM4_500	
		Verify Telemetry NSAU_MCHK SMMAN500	= FFFF <hex>	AND=SAM4_500	
		Verify Telemetry CHK_MCHK SMMBN500		AND=SAM4_500	
5.2.2		Verify checksum value			
		Check the received checksum against the expected value			
		Verify Telemetry CHK_MCHK SMMBN500	= expected value	AND=SAM4_500	
5.3		Command and verify the third checksum			

Load SPIRE OBS from ASW File: H_FCP_OBS_5110.xls Author: m.baker-hp	
---	--

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment
		Uplink the third SCM02500 memory check commands with ARM-GO			
5.3.1		Verify reception and contents of TM(6,10)			
		Note: A TM(6,10) packet will be received for each memory check command uplinked.			
5.3.1.1		IF SPIRE Prime			
		Verify Packet Reception Memory_Check_Absolute_Addresses Packet Mnemonic : SMEMCHK00500 APID : 1280 Type : 6 Subtype : 10 PI1 : PI2 :			
		Verify Telemetry MEMORYID_MCHK SMM8N500 = 0003 <hex>		AND=SAM4_500	
		Verify Telemetry STARTADDR_MCHK SMM9N500 = 5FFF <hex>		AND=SAM4_500	
		Verify Telemetry NSAU_MCHK SMMAN500 = 59 <hex>		AND=SAM4_500	
		Verify Telemetry CHK_MCHK SMMBN500		AND=SAM4_500	
5.3.1.2		ELSE SPIRE Redundant			
		Verify Packet Reception R_Memory_Check_Absolute_Addresses Packet Mnemonic : SMEMCHK00500 APID : 1281 Type : 6 Subtype : 10 PI1 : PI2 :			
		Verify Telemetry MEMORYID_MCHK SMM8N500 = 0003 <hex>		AND=SAM4_500	
		Verify Telemetry STARTADDR_MCHK SMM9N500 = 5FFF <hex>		AND=SAM4_500	
		Verify Telemetry NSAU_MCHK SMMAN500 = 59 <hex>		AND=SAM4_500	
		Verify Telemetry CHK_MCHK SMMBN500		AND=SAM4_500	
5.3.2		Verify checksum value			

Load SPIRE OBS from ASW File: H_FCP_OBS_5110.xls Author: m.baker-hp	 
---	--

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch	AIT Comment
		Check the received checksum against the expected value			
		Verify Telemetry CHK_MCHK SMMBN500	= expected value	AND=SAM4_500	
6		RETURN to calling proc. H_FCP_SPI_CLOM/H_FCP_SPI_CLOA		Next Step: END	
		Return to calling procedure H_FCP_SPI_CLOM or H_FCP_SPI_CLOA			
		End of Sequence			
End of Procedure					