Procedure Summary

Objectives

This procedure describes the steps needed to write, dump and check the selected Absolute addresses of the Safe Guard Memory, SGM.

A priori there is no need to write to SGM. Nevertheless this procedure can be used on a case-by-case basis for possible contingencies requiring to write to SGM.

Summary of Constraints

SGM is:
# loaded through TC(6,2);
# dumped through TC(6,5);
# checked through TC(6,9).

Read/Write operations in SGM are performed in tight sequence and ruled by a dedicated semaphore. No other particular precautions need to be taken. The logical addresses are to be used with TCs, the 16 MSBs being the MemoryID and the 16 LSBs the Start Address (32 bits aligned). All accesses to SGM memory must be 32-bit transfers, aligned to 32-bit boundaries.

These TCs will be delayed when there is an ongoing:
- TC(6,2) Load Memory
- TC(6,5) Dump Memory
- TC(6,9) Check Memory
- TC(8,4,1,1) Copy Memory

Spacecraft Configuration

Start of Procedure

n/a

End of Procedure

n/a

Reference File(s)

Input Command Sequences

Output Command Sequences

HRD3014B
HRD3014C
HRD3014D

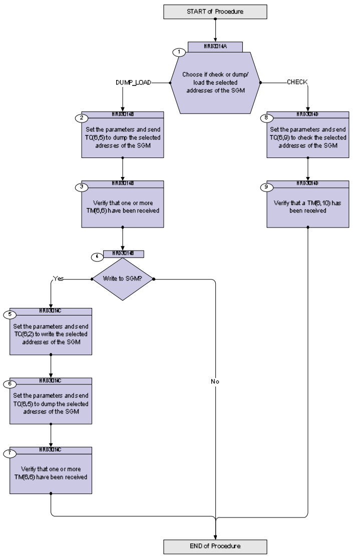
Referenced Displays

ANDs	**GRDs**	**SLDs**
		(None)

Configuration Control Information

DATE	FOP ISSUE	VERSION	MODIFICATION DESCRIPTION	AUTHOR	SPR REF
25/01/08	1	1	Created	cmevi-hp	
14/11/08	2	2	Procedure updated according to latest version received from industry on 12/09/2008	cmevi-hp	

Procedure Flowchart Overview


START of Procedure
HR03014A
1
Choose if check or dump/ load the selected addresses of the SGM
DUMP_LOAD
CHECK
HR03014B
2
Set the parameters and send TC(6,5) to dump the selected adresses of the SGM
HR03014B
3
Verify that one or more TM(6,6) have been received
HR03014B
4
Write to SGM?
Yes
No
HR03014C
5
Set the parameters and send TC(6,2) to write the selected addresses of the SGM
HR03014C
6
Set the parameters and send TC(6,5) to dump the selected adresses of the SGM
HR03014C
7
Verify that one or more TM(6,6) have been received
HR03014D
8
Set the parameters and send TC(6,9) to check the selected addresses of the SGM
HR03014D
9
Verify that a TM(6,10) has been received
END of Procedure

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch
		Beginning of Procedure		
		TC Seq. Name :HRD3014A (Dummy sequence) *TimeTag Type:* *Sub Schedule ID:* □		
1		*Choose if check or dump/load the selected addresses of the SGM*		Next Step: DUMP_LOAD 2 CHECK 8
1.1		*DUMP_LOAD*		□
1.2		*CHECK*		□
		TC Seq. Name :HRD3014B (Dump SGM addresses) *TimeTag Type:* *Sub Schedule ID:* □		
2		*Set the parameters and send TC(6,5) to dump the selected adresses of the SGM*		Next Step: 3
		When the CDMU receives this request it shall read the memory block, generate one or several TMs(6,6) containing the contents of this area and send them to the downlink.		
		In the TC(6,5) it is necessary to set the following parameters: **- Memory ID: identifier of the memory block of the on-board user from which data shall be dumped.** **- Start Address: start address (in SAUs, with the count starting from zero) within the memory block for dumping the data (see SGM allocation).** **- N: number of SAUs to be dumped.**		
		The fields Memory ID and Start Address are treated as one 32-bit field where the 16 least significant bits of the address is stored in Start Address and the 16 most significant bit in the Memory ID field.		
		The length Smallest Addressable Unit (SAU) is 8-bit. However all accesses to SGM memory must be 32-bit transfers, aligned to 32-bit boundaries.		

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch
		WARNING: **The specified address range must not span over several types of memory.** **Protected and unprotected parts of a memory are counted as separate memory types.** **SGM areas reserved for ASW and BSW are counted as separate memory types.**		
		SGMA Memory ID is 0x00B0. **SGMB Memory ID is 0x00E0.**		
		Execute Telecommand **DumpMem_AbsAddr** *Command Parameter(s) :* **Memory_ID DH003180** **Start_Address DH004180** **N DH105180** *TC Control Flags :* **GBM IL DSE** **--Y -- ---** *Subsch. ID : 10* Det. descr. : Dump Memory Using Absolute Addresses This Telecommand will not be included in the export	**DC602180** **Memory_ID** **Start_Address** **Number_of_SAUs**	
3		*Verify that one or more TM(6,6) have been received*		Next Step: 4
		Verify Packet Reception **Memory Dump - Absolute Addresses - SAU 8** *Packet Details:* **APID:** **Type:** **Subtype:** **PI1:** **PI2:**	**MemDmpAbsAdd** **16** **6** **6**	
		Each TM packet contains the following parameters:		
		Verify Telemetry **Memory_ID DE060180**		(None)
		Verify Telemetry **Start_Address DE061180**		(None)
		The Memory ID and Start Address identify the absolute memory address of the first reported SAU in the current TM packet. Thus can be different from the ones commanded in case several TM packets are required for the Dump.		
		Verify Telemetry **N DE062180**		(None)
		The following parameter is repeated N times		

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch
		Verify Telemetry **Dumped_Byte** **DE063180**		(None)
		Verify Telemetry **CheckSum** **DE064180**		(None)
4		*Write to SGM?*		Next Step: Yes 5 No END
		TC Seq. Name :HRD3014C (Write to SGM) *TimeTag Type:* *Sub Schedule ID:* □		
5		*Set the parameters and send TC(6,2) to write the selected addresses of the SGM*		Next Step: 6
		When the CDMU receives this Telecommand, it shall store the checksum of the received data, write the data block to the memory at the specified start address and re-read the memory area just written to, calculate and compare the checksum and report successful execution, TM(1,7), or an error report TM(1,8). In case of an error the loaded data shall be discarded.		
		In the TC(6,2) it is necessary to set the following parameters: **- Memory ID: identifier of the destination memory block.** **- Start Address: start address (in SAUs, with the count starting from zero) within the memory block for loading the data (see SGM allocation).** **- N: number of SAUs to be loaded. As the overall length of a TC packet cannot exceed 248 octets, the maximum length of the field "Data" will be 228 octets. N, expressed as number of SAUs, must be compatible with this boundary.** **- Data: repeated N times, data block to be loaded (in increasing order of SAU).** **- Checksum: CRC checksum that is used by the on-board user to verify the integrity of the data being loaded. This checksum is generated over the unpadded Data block to be loaded, (i.e. : excluding the optional spare octet) and is additional and different to the CRC word at the end of each packet.**		

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch
		The fields Memory ID and Start Address are treated as one 32-bit field where the 16 least significant bits of the address is stored in Start Address and the 16 most significant bit in the Memory ID field.		
		The length Smallest Addressable Unit (SAU) is 8-bit. However all accesses to SGM memory must be 32-bit transfers, aligned to 32-bit boundaries.		
		WARNING: **The specified address range must not span over several types of memory.** **Protected and unprotected parts of a memory are counted as separate memory types.** **SGM areas reserved for ASW and BSW are counted as separate memory types.**		
		WARNING: the following TC is a variable lenght TC therefore does not allow the definition of a generic procedure and it is intended to be just an example.		
		Execute Telecommand **LoadMem_AbsAdd_EvenByte** *Command Parameter(s) :* **Memory_ID** **DH003180** **Start_Address** **DH004180** **N** **DH005180** **Data** **DH006180** **Data** **DH006180** **Data** **DH006180** **Data** **DH006180** **CheckSum** **DH007180** *TC Control Flags :* **GBM IL DSE** **--Y -- ---** *Subsch. ID : 10* Det. descr. : Load Even number of bytes inMemory Using Absolute Addresses This Telecommand will not be included in the export	**DC601180** **Memory_ID** **Start_Address** **4 <hex>** **1st data byte** **2nd data byte** **3rd data byte** **4th data byte** **Checksum**	
		Note: DC600180 LoadMem_AbsAdd_OddByte should never be used as all accesses to SGM memory must be 32-bit transfers.		
6		*Set the parameters and send TC(6,5) to dump the selected adresses of the SGM*		Next Step: 7
		When the CDMU receives this request it shall read the memory block, generate one or several TMs(6,6) containing the contents of this area and send them to the downlink.		

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch
		In the TC(6,5) it is necessary to set the following parameters: **- <u>Memory ID:</u> identifier of the memory block of the on-board user from which data shall be dumped.** **- <u>Start Address:</u> start address (in SAUs, with the count starting from zero) within the memory block for dumping the data (see SGM allocation).** **- <u>N:</u> number of SAUs to be dumped.**		
		The fields Memory ID and Start Address are treated as one 32-bit field where the 16 least significant bits of the address is stored in Start Address and the 16 most significant bit in the Memory ID field.		
		The length Smallest Addressable Unit (SAU) is 8-bit. However all accesses to SGM memory must be 32-bit transfers, aligned to 32-bit boundaries.		
		<u>WARNING:</u> **The specified address range must not span over several types of memory.** **Protected and unprotected parts of a memory are counted as separate memory types.** **SGM areas reserved for ASW and BSW are counted as separate memory types.**		
		Execute Telecommand **DumpMem_AbsAddr** *Command Parameter(s) :* **Memory_ID DH003180** **Start_Address DH004180** **N DH105180** *TC Control Flags :* **GBM IL DSE** **--Y -- ---** *Subsch. ID : 10* Det. descr. : Dump Memory Using Absolute Addresses This Telecommand will not be included in the export	**DC602180** **Memory_ID** **Start_Address** **Number_of_SAUs**	
7		*Verify that one or more TM(6,6) have been received*		Next Step: END
		Verify Packet Reception **Memory Dump - Absolute Addresses - SAU 8** *Packet Details:* **APID:** **Type:** **Subtype:** **PI1:** **PI2:**	**MemDmpAbsAdd** **16** **6** **6**	
		Each TM packet contains the following parameters:		

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch
		Verify Telemetry **Memory_ID** **DE060180**		(None)
		Verify Telemetry **Start_Address** **DE061180**		(None)
		The Memory ID and Start Address identify the absolute memory address of the first reported SAU in the current TM packet. Thus can be different from the ones commanded in case several TM packets are required for the Dump.		
		Verify Telemetry **N** **DE062180**		(None)
		The following parameter is repeated N times		
		Verify Telemetry **Dumped_Byte** **DE063180**		(None)
		Verify Telemetry **CheckSum** **DE064180**		(None)
		TC Seq. Name :HRD3014D (Check addresses into) *TimeTag Type:* *Sub Schedule ID:* □		
8		*Set the parameters and send TC(6,9) to check the selected addresses of the SGM*		Next Step: 9
		When the CDMU receives this request it shall read and compute the checksum value of the indicated area of the memory using the CRC checksum algorithm. It then generates a report containing the checksum value computed.		
		In the TC(6,9) it is necessary to set the following parameters: **- <u>Memory ID:</u> identifier of the memory block of the on-board user which data shall be checked.** **- <u>Start Address:</u> start address (in SAUs, with the count starting from zero) within the memory block for dta to be checked (see SGM allocation).** **- <u>N:</u> number of SAUs on which the CRC checksum algorithm shall be applied.**		
		The fields Memory ID and Start Address are treated as one 32-bit field where the 16 least significant bits of the address is stored in Start Address and the 16 most significant bit in the Memory ID field.		

Step No.	Time	Activity/Remarks	TC/TLM	Display/ Branch
		The length Smallest Addressable Unit (SAU) is 8-bit. However all accesses to SGM memory must be 32-bit transfers, aligned to 32-bit boundaries.		
		WARNING: **The specified address range must not span over several types of memory.** **Protected and unprotected parts of a memory are counted as separate memory types.** **SGM areas reserved for ASW and BSW are counted as separate memory types.**		
		Execute Telecommand **ChkMem_AbsAdd** *Command Parameter(s) :* **Memory_ID** **DH003180** **Start_Address** **DH004180** **N** **DH105180** *TC Control Flags :* **GBM IL DSE** **--Y -- ---** *Subsch. ID : 10* Det. descr. : Check Memory Using Absolute Addresses This Telecommand will not be included in the export	**DC603180** **Memory_ID** **Start_Address** **Number_of_SAUs**	
9		*Verify that a TM(6,10) has been received*		Next Step: END
		Verify Packet Reception **Memory Check Report - Absolute Addresses** *Packet Details:* **APID:** **Type:** **Subtype:** **PI1:** **PI2:**	**MemChkRepAbs** **16** **6** **10**	

End of Procedure