HSO/FIRST-DPU

CARLO GAVAZZI SPACE SpA

Tipo Doc.: *Doc.Type:*	PLAN				N° DRD: *DRD N°:*	N.A.	
N° Doc.: *Doc. N°:*	**DPU-PL-CGS-002**	Ediz.: *Issue:* **1**		Data: *Date:* **5/04/2002**	Pagina *Page* **1**	Di *Of* **61**	
Titolo : *Title :*	**DPU-SW VERIFICATION AND VALIDATION PLAN/ACCEPTANCE TEST**						

	Nome & Funzione *Name & Function*	Firma *Signature*	Data *Date*	LISTA DI DISTRIBUZIONE *DISTRIBUTION LIST*	N	A	I
Preparato da: *Prepared by:*	FIRST-DPU TEAM			Interna / *Internal*			
				Legramandi S. (PA/QA)	1	X	
				Bertoli A. (DT/SW)	1	X	
				Longoni A. (DP/PL)	1	X	
					1	X	
Approvato da: *Approved by:*	Legramandi S. (PA/QA)						
	Di Gioia L. (PC/CC)						
	Bertoli A. (DT/SW)						
Applicazione autorizzata da: *Application authorized by:*	Longoni A. (DP/PL)			Esterna / *External*			
				Orfei R. (CNR-IFSI)	1	X	
Customer / Higher Level Contractor							
Accettato da: *Accepted by:*							
Approvato da: *Approved by:*				N=Numero di copie A=Applicazione I=Informazione *N=Number of copy A=Application I=Information*			

Gestione documenti: *Data Management:* --- Firma / *Signature* Data / *Date*	File: DPU-PL-CGS-002 is 1.doc

REGISTRAZIONE DELLE MODIFICHE / *CHANGE RECORD*			
EDIZIONE *ISSUE*	DATA *DATE*	AUTORIZZAZIONE *CHANGE AUTHORITY*	OGGETTO DELLA MODIFICA E SEZIONI AFFETTE *REASON FOR CHANGE AND AFFECTED SECTIONS*
1	5/04/2002		

LISTA DELLE PAGINE VALIDE / *LIST OF VALID PAGES*									
PAGINA *PAGE*	EDIZIONE *ISSUE*	PAGINA *PAGE*	EDIZIONE *ISSUE*	PAGINA *PAGE*	EDIZIONE *ISSUE*	PAGINA *PAGE*	EDIZIONE *ISSUE*	PAGINA *PAGE*	EDIZIONE *ISSUE*
1 - 59	1								

TABLE OF CONTENT

		N° Doc: *Doc N°:*	**DPU-PL-CGS-002**
HSO/FIRST-DPU		Ediz.: *Issue:* **1**	Data: *Date:* **5/04/2002**
DPU-SW VERIFICATION AND VALIDATION PLAN/ACCEPTANCE TEST		Pagina *Page* **7**	di *of* **61**

CARLO GAVAZZI SPACE SpA

1. INTRODUCTION.

1.1 PURPOSE.

This document is relevant to the HSO/FIRST DPU project, particularly it refers to the development of the DPU BASIC SW product.

The purpose of the document is to define and to describe the test activities to be performed during the DPU BASIC SW acceptance test (SVVP/AT).

The DPU BASIC SW is verified against the [AD 9] concerning the applicable requirements, [AD 10] and the [AD 11] documents.

The scope of acceptance testing is to verify the compliance of DPU BASIC S/W with the user requirements, as stated in the URD documents. The Input to the Acceptance test is the DPU BASIC SW tested at system level and the output will be the DPU BASIC SW Accepted Software.

1.2   DOCUMENT STRUCTURE.

For the document structure, the table of contents reflects the recommendation of ESA reported in [RD1].
The SVVP/AT content is splitted into four main sections:
Section 1,2,3 are introduction sections.
The section 4 details the Test Activities and it is divided in five sections:

❑   Section 4.1: Test Plan. This section list the software items and the software features to be tested. The sections details the test activities to carry out the test and the External Environment to be set for test support.

The following sections will be completed.
Section 4.2:Test Design.
Section 4.3:Test Cases.
Section 4.4:Test Procedures.
Section 4.5:Test Results.

2. RELEVANT DOCUMENTS.

2.1 APPLICABLE DOCUMENTS.

AD	Doc.Number	Issue/Date	Rev.	Title/Applicability
1	CNR.IFSI.2000TR01	1 Sep 15, 2000		Documento di Specifiche Tecniche per il Contratto delle Data Processing Units del Satellite First dell'ESA" IFSI
2	DPU-SP-CGS-001	2 Oct 30, 2000		FIRST CPU Board Specification
3	DPU-SP-CGS-002	1 Oct 12, 2000		Payload & Spacecraft Interface Board Specification
4		Nov 30, 1999		Technical proposal CGS (Ref. S9-030 November 99)
5				Allegato Tecnico al Contratti ASI
6	CNR.IFSI.2001TR02	1 April 5, 2001		Herschel Space Observatory DPU Applicable Documents Guidelines
7	SCI-PT-ICD-7527	1 Sep 1, 2000		Packet Structure - Interface Control Document
8	IFSI-OBS-PL-2000	1 Oct 13, 2000		DPU/ICU OBS PA Plan
9	IFSI-OBS-SP-2000-001	1		DPU/ICU OBS URD, On Board S/W User Requirement Document
10	CNR.IFSI.2001TR01	Draft3 March 21, 2001		DPU/ICU Switch-ON Procedure
11	DPU-SQ-CGS-001	2 July 12, 2001		HSO/FIRST Software Requirement Document

2.2 REFERENCE DOCUMENTS.

RD #	Doc Number	Issue	Date	Rev	Date	Title
1	ESA PSS-05-0	2	Feb, 1991			ESA Software Engineering standards

3. DEFINITIONS AND ACRONYMS.

- AD   Architectural Design
- ADD   Architectural Design Document
- API   Application Program Interface
- ASW   Application Software
- AT   Acceptance Test
- CNR   Consiglio Nazionale Ricerche
- CPU   Control Processing Unit
- CDMS   Computer Data Management System
- CGS   Carlo Gavazzi Space
- COTS   Commercial Of The Shelf
- CSPEC   Control Specification
- DD   Detail Design
- DDD   Detailed Design Document
- DFD   Data Flow Diagram
- DM   Data memory
- EEPROM   Electrical Erasable Programmable Read Only Memory
- ESA   European Space Agency
- GUI   Graphic User Interface
- HK   House-Keeping
- HW   Hardware
- IFSI   Istituto di Fisica dello Spazio Interplanetario
- IT   Integration Test
- MMI   Man Machine Interface
- OBS   On board software
- PAT   Process Activation Table
- PM   Program Memory
- PROM   Programmable Read Only Memory
- PSPEC   Processor Specification
- RAM   Random Access Memory
- S/W   Software
- SDE   Software Development Environment
- SR   Software Requirements
- SRD   Software Requirement Document
- ST   System Test
- STD   State Transition Diagram
- SVVP   Software Validation and Verification Plan
- TBC   To be Confirmed
- TBD   To be Defined
- TE   Test Equipment
- UT   Unit Test

4. TEST ACTIVITIES.

4.1   ACCEPTANCE TEST PLAN.

This plan defines the "Acceptance Test" activities to be performed on the DPU BASIC SW. It covers the Acceptance tests with respect to the requirements in the URD's documents. Only a subset of overall user requirements are applicable to the DPU BASIC S/W; the first step is to identify the applicable UR's subset. The picture shows the process used to extract the subset of UR's that generates the test cases for acceptance test. The Software Requirements are traced on three high level documents : [AD 9], [AD 10], [AD 7]. From the SRD forward traceability matrix will be built the backward traceability matrix. This matrix will provide the applicable UR's subset to the DPU BASIC SW. The Acceptance test will cover all the UR's identified.
The DPU-BASIC SW "Acceptance Tests" will be performed on the DPU EM, QM and FM models.
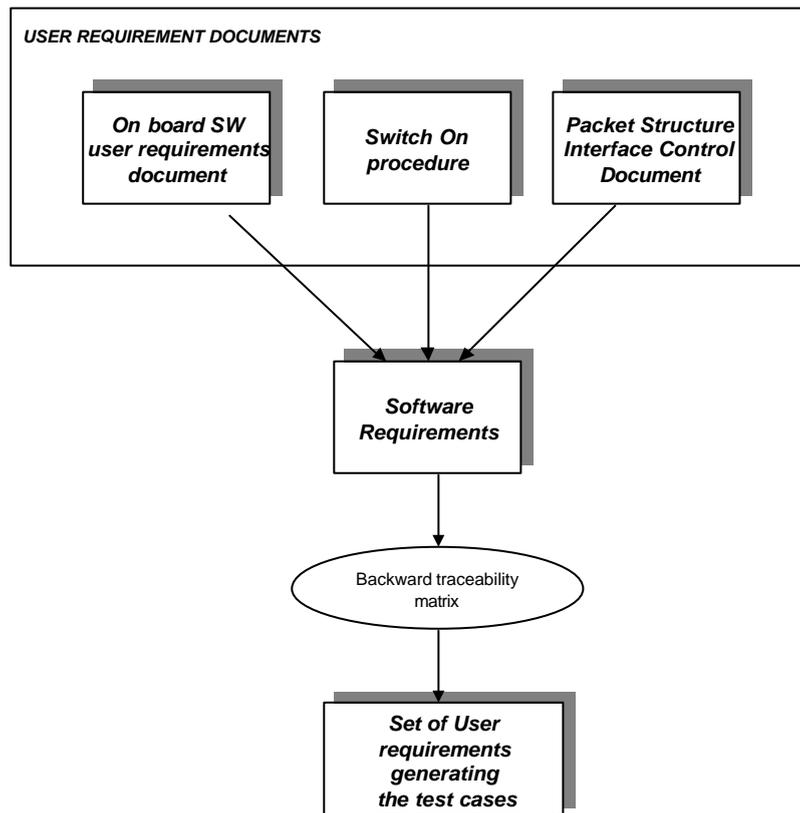
Figure 4-1: UR's identification process

4.2   TEST ITEMS.

The item to be tested is the DPU BASIC SW. This is composed of two parts as follow:
- DPU BOOT S/W;
- DPU DRIVERS S/W;

The DPU BOOT S/W is the Software in PROM and it is responsible for:
- the uploading of the Application Program from EEPROM to Program Memory;
- the uploading of the Application Software via MIL-STD-1553B from the external to the Program Memory;

The DPU BOOT S/W concludes its task running the uploaded application software.
The DPU DRIVERS S/W are the software drivers hiding the H/W interfaces of the DPU board devices.
They include the drivers of the following board devices:
- MIL-STD-1553B;
- IEEE1355;
- WATCHDOG;
- EEPROM;

The DPU BASIC S/W will be tested as two functional independent items. In a first phases will be tested the DPU BOOT SW while in a second phases will be tested the DPU DRIVERS S/W.
The DPU BOOT S/W will use for testing an "Application Trial Program" composed of all the MIL-STD 1553B Software Drivers, Watchdog Drivers, Virtuoso Kernel, and a Stub program for generating a simple message via MILBUS informing the good operation of the uploaded program. The test will verify that the "Application Trial Program" will correctly be uploaded either from EEPROM or via MIL-STD-1553B when a one of this failure DM fail, PM Fail or EEPROM occur. After the DPU BOOT S/W program conclusion, the "Application Trial Program" operation will validate the DPU BOOT S/W.
The DPU DRIVERS SW will be linked to an "Application Test Program". For the uploading of the "Application Test Program" in Program Memory will be exploited the DPU BOOT. The "Application Test Program" will be composed of the following Software items:
- The DPU DRIVERS SW for testing;
- A stub program supporting the test of the DPU DRIVERS S/W;
- Virtuoso Kernel.

The "Application test program" will be able to call the Software Drivers functions receiving suitable command via MILBUS and reporting the results using the MILBUS link.

4.3   FEATURES TO BE TESTED.

All the applicable User Requirements to DPU BASIC SW , as defined in UR's identification process will be verified.
The verification of the UR requirements is done:
1. by <u>functional test</u> , i.e. by using a test procedure and comparing expected results with obtained results.
2. by <u>review of design</u>, i.e. by verifying that the requirement is implemented in the Software requirements Document and Architectural and Detailed Design Document and in general, in the SW documentation.

The verification method is extrapolated on the Verification Matrix included in the Software Requirements Document.
For all verifications the relevant acceptance procedure will be described in the Test Procedure.

4.4   FEATURES NOT TO BE TESTED.

TBD

4.5   APPROACH.

In order to test the overall DPU BASIC SW two approaches will be considered. The DPU BASIC SW requirements can be divided in two main groups:
- DPU BOOT S/W requirements;
- DPU DRIVERS S/W requirements.

This two groups of requirements can be tested with different test philosophy. The Test of DPU BASIC S/W will follow a pre determinate order as follow:
1) DPU BOOT S/W;
2) DPU DRIVERS S/W exploiting the DPU BOOT S/W during the uploading phase.

4.5.1   DPU BOOT S/W APPROACH.

The DPU BOOT S/W can be considered a standalone Software with homogeneous external interfaces, in this case the interface is the MIL-STD-1553B. The Test Environment will provide the MIL-STD-1553B link and interface.
In order to carry out the DPU BOOT S/W test the following major tools/activities are identified.
- Tools to be provided;
    - ADSP21020 Software Development Environment to support the S/W Stub item development;
    - Software tools providing MMI (GUI) to the MIL-STD-1553B. The tool will allow the two functions as follow:
        - Transmission of Telecommands packets in order to transfer the "Application Test Program" to the DPU board;
        - Reception of Telemetry Packet containing Event Report messages from the DPU board;
- Activities to performed;
    - A Software for "Application Trial Program" segmentation will be developed;
      The Software includes two functionalities:
        - Generation of "Segmented Application Trial Program for EEPROM programming";
        - Generation of "Segmented Application Trial Program for MIL BUS transmission";
      This Software will be used during the Test preparation; a relevant set of "Segmented Application Trial Program" will be generated in order to cover all the identified test cases.
    - An "Application Trial Program" will be developed in order to dump via MIL-STD-1553B the Program Memory contained on request, when the DPU BOOT S/W will have uploaded the "Application Trial Program". The "Application Trial Program" will download MILBUS messages informing the Tester of the program execution after the DPU BASIC S/W program conclusion.
    - Condor Bustools configuration files for setting the MIL-STD-1553B transmit and receive packets.
    - Test Procedure development;
    - Test Procedure execution;
    - Test Reports;

4.5.2   DPU DRIVERS S/W APPROACH.

The DPU DRIVERS S/W will be object or library to be linked to an "Application Program" that will be developed by the customer.
In this case the interface will not be homogeneous but the DPU DRIVERS S/W will have from one side physical communication interfaces and from an other side Software Interfaces to the Application Software. The DPU TE will provide all the physical communication interfaces but will not be able to

provide the Software Interfaces. A S/W Stub item is required in order to provide to the DPU DRIVERS S/W all the boundary conditions. The S/W Stub item will have the following capabilities:

❑ The S/W Stub will be linked to the following software items in order to generate the "Application Test Program".
▪ DPU DRIVERS S/W;
▪ Virtuoso Kernel;
❑ The S/W Stub item will be able to call all the Application Interface Functions included in the DPU DRIVERS S/W item.
❑ The S/W Stub item will be able to receive customized test command via MIL-STD-1553B for calling the API functions included in the DPU DRIVERS S/W.
❑ The S/W Stub item will be able to receive customized test command via MIL-STD-1553B for calling S/W stub services functions. The S/W stub service functions are TBD.
❑ The S/W Stub item will be able to receive customized test command via IEEE1355 Spacewire for calling the API functions included in the DPU DRIVERS S/W (TBC).
❑ The S/W Stub item will be able to receive customized test command via IEEE1355 Spacewire for calling S/W stub services functions (TBC).
❑ The S/W Stub item will reserve a dedicated Data Memory area for each API function of the DPU DRIVERS S/W in order to upload the functions parameters and the functions results.
❑ The S/W Stub item will be able to download the Data Memory area relevant to each API function.
❑ The S/W Stub item will be able to download specified Data Memory and Program Memory areas.
❑ The S/W Stub item will be able to download specified Parameters or Data.
❑ The S/W Stub item will composed of one or more simple S/W stub.
❑ The S/W Stub item will be able to send the echo of the received command.

The S/W Stub program source code in sheet format will be attached to the Test Procedures in order to document the Testing.

The "Application Test Program" will be uploaded in Program Memory and executed using the DPU BOOT S/W.

In order to carry out the DPU DRIVERS S/W test, the following major tools/activities are identified.

▪ Tools to be provided:
  ▪ Software tools providing MMI (GUI) to the MIL-STD-1553B board. The tool will allow the two function as follow:
    ▪ Sending of command to the "Application Test Program";
    ▪ Receiving of Downloaded Data from "Application Test Program";
  ▪ Software tools providing MMI to the IEEE1355 Spacewire board. The tool will allow two function as follow:
    ▪ Sending of command to "Application Test Program";
    ▪ Receiving of Downloaded Data from "Application Test program";
  ▪ ADSP21020 Software Development Environment to support the S/W Stub item development;
Activities to be performed:
▪ An "Application Test Program" will be developed and in particular the S/W Stub item for supporting the DPU DRIVERS S/W testing.
▪ Condor Bustools configuration files for setting the MIL-STD-1553B transmit and receive packets.
▪ Test Procedure Development.
▪ Test Report.

4.6   ITEM PASS/FAIL CRITERIA.

Each test case specifies the relevant pass/fail criteria.

4.7   RESUMPTION CRITERIA.

TBD.

4.8   TEST DELIVERABLES.

Before testing begins, the following items will be delivered:
- Test Plan / Acceptance Test;
- Test Design (TBC);
- Test Cases (TBC);
- Test Procedure;

At the End of testing, the following items will be delivered
- Test reports;
- Problem Reports;

A "Test Report" will be prepared, collecting all the "test input commands" and "test output results". The record of acceptance tests (with customer signature) will be included in the DPU-BASIC SW Test Reports.

Anomaly reporting and resolution is performed by means of Software Problem Reports, Software Change Requests and Software Modification Reports, according to procedure TBD of the PA Plan [AD 8]

4.9   TESTING TASKS.

1) The following task are identified in order to prepare and perform for DPU BOOT S/W testing:
   - The DPU PROM will be programmed uploading the DPU BOOT S/W.
   - The "Application Trial Program" will be prepared. It will include the following S/W items:
     - MIL-STD 1553 S/W Drivers;
     - Watchdog Drivers;
     - Stub program to support the testing;
     - Virtuoso Kernel.
   - A predefined set of "Segmented Application Trial Program" will be generated using the Software for the "Application Program" segmentation. In this phase will be generated two type of "Segmented Application Test Program" sets, as follow:
     - Segmented Application Test Program for EEPROM programming;
     - Segmented Application Test Program for uploading via MIL-STD-1553B.
   - The EEPROM will be programmed using a selected "Segmented Application Test Program for EEPROM programming" on the base of the relevant test cases
   - A set of "Segmented Application Test Program for uploading via MIL-STD-1553B" will be copied on the DPU TE ready to be sent to the DPU board via MIL-STD-1553B using the condor Mil bus.
   - The DPU BOOT S/W testing will start.
2) The following task are identified in order to prepare for and perform The DPU DRIVER S/W testing:
   - The DPU BOOT S/W testing has already .been done.
   - An "Application Test Program" will be prepared. It will include the following S/W items:
     - DPU DRIVERS S/W;
     - Stub program to support the testing;
     - Virtuoso Kernel;
   - A predefined "Segmented Application Test Program for EEPROM programming" will be generated using the Software for the "Application Program" segmentation.
   - The EEPROM will be programmed using the "Segmented Application Test Program for EEPROM programming".
   - The DPU DRIVERS S/W test will start.

4.10 ENVIRONMENTAL NEEDS.

For the Acceptance Test a DPU Test Equipment (TE) and the DPU SDE is required (TBC).
The DPU TE is composed of the following H/W and S/W items:
- Hardware:
  - 1 PC Pentium III 350 Mhz, 128Mbyte RAM, Hard disk 4GigaBye, 2 slots PCI min, Monitor 17 inches,
  - Condor MIL-STD-1553B PCI board;
  - IEEE1355 Spacewire 4 Links PCI board.
- Software:
  - Condor BusTools Software;
  - IEEE1355 S/W Test;
  - Windows2000 Operative System;
  - Software for generating "Segmented Application Test Program" ready to be uploaded from EEPROM or via MIL-STD-1553B;

The DPU SDE is composed of the following H/W and S/W items:
- Hardware:
  - Minimum requirements: PC Pentium II 233 Mhz, 128Mbyte RAM, Hard disk 4GigaBye, Monitor 17 inches;
  - ADSP21020 JTAG Emulator;
- Software:
  - Window2000 Operative system;
  - ADSP21020 Software Development Environment and ADSP21020 emulator;
  - Virtuoso 4.2 (TBC);

The DPU TE and DPU SDE should communicate using a network.

4.11 STAFFING AND TRAINING NEEDS.

TBD

4.12 SCHEDULE.

The schedule of testing activities is reported in the HSO/FIRST-DPU Project Plan, as a part of the overall schedule.

4.13 RISKS AND CONTINGENCIES.

None.

4.14 APPROVALS.

Approvals are shown on the cover page of the plan.

5. TEST DESIGNS.

5.1 DPU BOOT S/W TEST DESIGN.

1	Test Design Identifier	TD-BOOTSW
2	Features to be tested	Functional verification of boot procedure.
3	Approach refinement	
4	Test Case Identification	010 – High Level Requirement test and PM Test. 020 - DM Test. 030 – EEPROM Test. 040 – EEPROM to PM Load test. 050 - DM to PM load Test.
5	Feature pass/fail criteria	

5.2 1355 DRIVER TEST DESIGN.

1	Test Design Identifier	TD-DRV1553
2	Features to be tested	Functional verification of 1355 Drivers.
3	Approach refinement	
4	Test Case Identification	010 –1355 Power On. 020 – Reset Link. 030 – Time-out. 040 – Open Link. 050 – Close Link. 060 – Start Link as Master. 070 – Start Link as Slave. 080 – Stop Link. 090 – Get Status Register. 100 – Write Link. 110 – Read Link. 120 – Get Link status. 130 – Read Packets. 140 – Get Last Read Size. 150 – Get Last Write Size. 160 – Get Last Packet Number. 170 – Write to Board Memory. 180 – Read from board memory. 190 – Write Register. 200 – Read Register. 210 – Get Link State.
5	Feature pass/fail criteria	

5.3   1553 DRIVER TEST DESIGN.

1	Test Design Identifier	TD-DRV1553
2	Features to be tested	Functional verification of 1355 Drivers.
3	Approach refinement	
4	Test Case Identification	010 –1553 Power On and DDC chip configuration. 020 – Tx 1553 SA Messages 030 – Rx 1553 SA Messages 040 – Close 1553 and Structures de-allocation
5	Feature pass/fail criteria	

5.4   EEPROM DRIVER TEST DESIGN.

1	Test Design Identifier	TD-DRV1553
2	Features to be tested	Functional verification of 1355 Drivers.
3	Approach refinement	
4	Test Case Identification	010 –
5	Feature pass/fail criteria	

5.5   WATCHDOG TEST DESIGN.

1	Test Design Identifier	TD-DRVWD
2	Features to be tested	Functional verification of Watchdog driver.
3	Approach refinement	
4	Test Case Identification	010 –
5	Feature pass/fail criteria	

6. TEST CASE SPECIFICATIONS.

6.1 DPU BOOT S/W TEST CASE SPECIFICATIONS.

6.1.1 BOOT S/W HIGH LEVEL REQUIREMENT TEST.

1	Test Design Identifier	TD-BOOTSW -010
2	Test Items	High Level Requirement and PM test.
3	Input Specifications	[AD 11] SRD-3.1.1.0-000 [AD11] SRD-3.1.1.0-010 [AD11] SRD-3.1.1.0-020 [AD11] SRD-3.1.1.0-030 [AD11] SRD-3.1.1.0-150 [AD11] SRD-3.1.1.0-160 [AD11] SRD-3.1.1.0-170 [AD11] SRD-3.1.1.0-180 [AD11] SRD-3.1.1.0-190 [AD11] SRD-3.1.1.0-200 [AD11] SRD-3.1.1.0-210 [AD11] SRD-3.1.1.0-220 [AD11] SRD-3.1.1.0-230 [AD11] SRD-3.1.1.0-250 [AD11] SRD-3.1.1.0-260 [AD11] SRD-3.1.1.0-270 [AD11] SRD-3.1.1.0-280 [AD11] SRD-3.1.1.0-290 [AD11] SRD-3.1.1.0-300 [AD11] SRD-3.1.1.0-310 [AD11] SRD-3.1.1.0-320 [AD11] SRD-3.1.1.0-350 [AD11] SRD-3.1.1.0-360 [AD11] SRD-3.1.1.0-370 [AD11] SRD-3.1.5.0-000 [AD11] SRD-3.1.5.0-010 [AD11] SRD-3.1.5.0-020 [AD11] SRD-3.1.5.0-030 [AD11] SRD-3.1.5.0-040 [AD11] SRD-3.1.1.0-050 [AD11] SRD-3.1.1.0-060 [AD11] SRD-3.1.1.0-070 [AD11] SRD-3.1.1.0-080 [AD11] SRD-3.1.1.0-090 [AD11] SRD-3.1.1.0-100 [AD11] SRD-3.1.1.0-110 [AD11] SRD-3.1.1.0-120 [AD11] SRD-3.1.1.0-130 [AD11] SRD-3.1.1.0-140
4	Output Specifications	[AD7] Event reporting section.
5	Environmental needs	DSP emulator + PC with Condor MIL-STD-1553B PCI board and Bus tool SFW.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message that has to be done by AIV people.
7	Interface Dependencies	

6.1.2 DM TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-BOOTSW -020
2	Test Items	Data Memory
3	Input Specifications	[AD11] SRD-3.1.4.0-000 [AD11] SRD-3.1.4.0-010 [AD11] SRD-3.1.4.0-020 [AD11] SRD-3.1.4.0-030 [AD11] SRD-3.1.4.0-040 [AD11] SRD-3.1.1.0 Overall
4	Output Specifications	[AD7] Event reporting section.
5	Environmental needs	DSP emulator + PC with Condor MIL-STD-1553B PCI board and Bus tool SFW.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message that has to be done by AIV people.
7	Interface Dependencies	

6.1.3 PM TEST CASE SPECIFICATION

1	Test Design Identifier	TD-BOOTSW -025
2	Test Items	Program Memory
3	Input Specifications	[AD11] SRD-3.1.5.0-010 [AD11] SRD-3.1.5.0-020 [AD11] SRD-3.1.5.0-030 [AD11] SRD-3.1.5.0-040 [AD11] SRD-3.1.1.0 Overall
4	Output Specifications	[AD7] Event reporting section.
5	Environmental needs	DSP emulator + PC with Condor MIL-STD-1553B PCI board and Bus tool SFW.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message that has to be done by AIV people.
7	Interface Dependencies	

6.1.4 EEPROM TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-BOOTSW –030
2	Test Items	EEPROM
3	Input Specifications	[AD11] SRD-3.1.1.0-050 [AD11] SRD-3.1.6.0-000 [AD11] SRD-3.1.6.0-010 [AD11] SRD-3.1.6.0-020 [AD11] SRD-3.1.6.0-030 [AD11] SRD-3.1.6.0-040 [AD11] SRD-3.1.6.0-050 [AD11] SRD-3.1.6.0-060 [AD11] SRD-3.1.6.0-070 [AD11] SRD-3.1.1.0 Overall
4	Output Specifications	[AD7] Event reporting section.
5	Environmental needs	DSP emulator + PC with Condor MIL-STD-1553B PCI board and Bus tool SFW.

6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message that has to be done by AIV people.
7	Interface Dependencies	

6.1.5 EEPROM TO PM LOAD TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-BOOTSW –040
2	Test Items	EEPROM to PM Load
3	Input Specifications	[AD11] SRD-3.1.7.0-000 [AD11] SRD-3.1.7.0-010 [AD11] SRD-3.1.7.0-020 [AD11] SRD-3.1.7.0-030 [AD11] SRD-3.1.7.0-040
4	Output Specifications	[AD7] Event reporting section.
5	Environmental needs	DSP emulator + PC with Condor MIL-STD-1553B PCI board and Bus tool SFW.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message that has to be done by AIV people.
7	Interface Dependencies	

6.1.6 DM TO PM LOAD TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-BOOTSW –050
2	Test Items	DM to PM Load
3	Input Specifications	[AD11] SRD-3.1.8.0-000 [AD11] SRD-3.1.8.0-010 [AD11] SRD-3.1.8.0-020 [AD11] SRD-3.1.8.0-030 [AD11] SRD-3.1.8.0-040
4	Output Specifications	[AD7] Event reporting section.
5	Environmental needs	DSP emulator + PC with Condor MIL-STD-1553B PCI board and Bus tool SFW.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message that has to be done by AIV people.
7	Interface Dependencies	

	HSO/FIRST-DPU	N° Doc: *Doc N°:*	**DPU-PL-CGS-002**
		Ediz.: *Issue:* **1**	Data: *Date:* **5/04/2002**
	DPU-SW VERIFICATION AND VALIDATION PLAN/ACCEPTANCE TEST	Pagina *Page* **22**	di *of* **61**

CARLO GAVAZZI SPACE SpA

6.2   1355 DRIVER TEST CASE SPECIFICATIONS.

6.2.1   1355 POWER ON TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –010
2	Test Items	Power ON Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.2   RESET LINK TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –020
2	Test Items	Reset Link Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-010 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.3   TIME-OUT TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –030
2	Test Items	Time-out.
3	Input Specifications	[AD11] SRD-3.2.0.0-030 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.4   OPEN LINK TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –040
2	Test Items	Open Link Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-030 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.5   CLOSE LINK TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –050
2	Test Items	Close Link Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-040 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.6   START LINK AS MASTER TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –060
2	Test Items	Start Link as Master Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-050 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.7   START LINK AS SLAVE TEST CASE SPECIFICAT ION.

1	Test Design Identifier	TD-DV1355 –070
2	Test Items	Start Link as Slave Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-060 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool

		SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.8 STOP LINK TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –080
2	Test Items	Write Link Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-070 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.

6.2.9 GET STATUS REGISTER TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –090
2	Test Items	Get Status Register Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-080 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.10 WRITE LINK TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –100
2	Test Items	Write Link Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-090 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.11 READ LINK TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –110
2	Test Items	Read Link Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-120 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.12 GET LINK STATUS TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –120
2	Test Items	Get Link Status Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-140 [AD11] SRD-3.2.0.0-150 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.13 READ PACKET TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –130
2	Test Items	Read Packet Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-120 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.14 GET LAST READ SIZE TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –140
2	Test Items	Get Last Read Size Packet Function.

3	Input Specifications	[AD11] SRD-3.2.0.0-170 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.15 GET LAST WRITE SIZE TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –150
2	Test Items	Get Last Write Size Packet Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-190 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.16 GET LAST PACKET NUMBER TEST CASE.

1	Test Design Identifier	TD-DV1355 –160
2	Test Items	Get Last Packet Number Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-180 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.17 WRITE TO BOARD MEMORY.

1	Test Design Identifier	TD-DV1355 –170
2	Test Items	Write to Board memory Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-210 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.

6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.18 READ FROM BOARD MEMORY TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –180
2	Test Items	Read From Board Memory Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-210 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.19 WRITE REGISTER TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –190
2	Test Items	Write Register Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-200 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.20 READ REGISTER TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –200
2	Test Items	Read Register Function.
3	Input Specifications	[AD11] SRD-3.2.0.0-200 [AD11] SRD-6.1.0.0-000
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.2.21 GET LINK STATE TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1355 –210
2	Test Items	Get Link State Function.
3	Input Specifications	AD11] SRD-3.2.0.0-200
4	Output Specifications	
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool SFW + 4-links Spacewire board and 4 links application software.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and 1355 message that has to be done by AIV people.
7	Interface Dependencies	

6.3   1553 DRIVER TEST CASE SPECIFICATIONS.

6.3.1   OPEN MIL-STD 1553 CHANNEL TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DV1553 –010
2	Test Items	1553 Drivers
3	Input Specifications	[AD11] SRD-3.3.2.0-000
		[AD11] SRD-3.3.2.0-010
		[AD11] SRD-3.3.3.1-000
		[AD11] SRD-3.3.3.1-010
		[AD11] SRD-3.3.3.1-020
		[AD11] SRD-3.3.3.1-030
		[AD11] SRD-3.3.3.1-040
		[AD11] SRD-3.3.3.1-050
		[AD11] SRD-3.3.3.1-060
		[AD11] SRD-3.3.3.2-000
		[AD11] SRD-3.3.3.2-010
		[AD11] SRD-3.3.3.2-020
		[AD11] SRD-3.3.3.2-030
		[AD11] SRD-3.3.3.2-040
		[AD11] SRD-3.3.3.2-050
		[AD11] SRD-3.3.3.2-060
		[AD11] SRD-3.3.3.3-000
		[AD11] SRD-3.3.3.3-010
		[AD11] SRD-3.3.3.3-020
		[AD11] SRD-3.3.3.3-030
		[AD11] SRD-3.3.3.3-040
		[AD11] SRD-3.3.3.3-050
		[AD11] SRD-3.3.3.3-060
		[AD11] SRD-3.3.3.3-070
		[AD11] SRD-3.3.3.3-080
		[AD11] SRD-3.3.3.3-090
		[AD11] SRD-3.3.3.3-100
		[AD11] SRD-3.3.3.3-110
		[AD11] SRD-3.3.3.3-120
		[AD11] SRD-3.3.3.3-130
		[AD11] SRD-3.3.3.3-140
		[AD11] SRD-3.3.3.3-150
		[AD11] SRD-3.3.3.3-160
4	Output Specifications	[AD7] Event reporting section.
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool S/W and ADSP emulator.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and DDC look up table that has to be done by AIV people.
7	Interface Dependencies	

6.3.2 TX MIL-STD 1553 MESSAGES

1	Test Design Identifier	TD-DV1553 –020
2	Test Items	1553 Drivers
3	Input Specifications	[AD11] SRD-3.3.2.0-000
		[AD11] SRD-3.3.2.0-010
		[AD11] SRD-3.3.3.1-000
		[AD11] SRD-3.3.3.1-010
		[AD11] SRD-3.3.3.1-020
		[AD11] SRD-3.3.3.1-030
		[AD11] SRD-3.3.3.1-040
		[AD11] SRD-3.3.3.1-050
		[AD11] SRD-3.3.3.1-060
		[AD11] SRD-3.3.3.2-000
		[AD11] SRD-3.3.3.2-010
		[AD11] SRD-3.3.3.2-020
		[AD11] SRD-3.3.3.2-030
		[AD11] SRD-3.3.3.2-040
		[AD11] SRD-3.3.3.2-050
		[AD11] SRD-3.3.3.2-060
		[AD11] SRD-3.3.3.3-000
		[AD11] SRD-3.3.3.3-010
		[AD11] SRD-3.3.3.3-020
		[AD11] SRD-3.3.3.3-030
		[AD11] SRD-3.3.3.3-040
		[AD11] SRD-3.3.3.3-050
		[AD11] SRD-3.3.3.3-060
		[AD11] SRD-3.3.3.3-070
		[AD11] SRD-3.3.3.3-080
		[AD11] SRD-3.3.3.3-090
		[AD11] SRD-3.3.3.3-100
		[AD11] SRD-3.3.3.3-110
		[AD11] SRD-3.3.3.3-120
		[AD11] SRD-3.3.3.3-130
		[AD11] SRD-3.3.3.3-140
		[AD11] SRD-3.3.3.3-150
		[AD11] SRD-3.3.3.3-160
4	Output Specifications	[AD7] Event reporting section.
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool S/W and ADSP emulator.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and DDC chip lookup table that has to be done by AIV people.
7	Interface Dependencies	

6.3.3   RX MIL-STD 1553 MESSAGES

1	Test Design Identifier	TD-DV1553 –030
2	Test Items	1553 Drivers
3	Input Specifications	[AD11] SRD-3.3.2.0-000
		[AD11] SRD-3.3.2.0-010
		[AD11] SRD-3.3.3.1-000
		[AD11] SRD-3.3.3.1-010
		[AD11] SRD-3.3.3.1-020
		[AD11] SRD-3.3.3.1-030
		[AD11] SRD-3.3.3.1-040
		[AD11] SRD-3.3.3.1-050
		[AD11] SRD-3.3.3.1-060
		[AD11] SRD-3.3.3.2-000
		[AD11] SRD-3.3.3.2-010
		[AD11] SRD-3.3.3.2-020
		[AD11] SRD-3.3.3.2-030
		[AD11] SRD-3.3.3.2-040
		[AD11] SRD-3.3.3.2-050
		[AD11] SRD-3.3.3.2-060
		[AD11] SRD-3.3.3.3-000
		[AD11] SRD-3.3.3.3-010
		[AD11] SRD-3.3.3.3-020
		[AD11] SRD-3.3.3.3-030
		[AD11] SRD-3.3.3.3-040
		[AD11] SRD-3.3.3.3-050
		[AD11] SRD-3.3.3.3-060
		[AD11] SRD-3.3.3.3-070
		[AD11] SRD-3.3.3.3-080
		[AD11] SRD-3.3.3.3-090
		[AD11] SRD-3.3.3.3-100
		[AD11] SRD-3.3.3.3-110
		[AD11] SRD-3.3.3.3-120
		[AD11] SRD-3.3.3.3-130
		[AD11] SRD-3.3.3.3-140
		[AD11] SRD-3.3.3.3-150
		[AD11] SRD-3.3.3.3-160
4	Output Specifications	[AD7] Event reporting section.
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool S/W and ADSP emulator.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and DDC chip lookup table that has to be done by AIV people.
7	Interface Dependencies	

6.3.4   CLOSE MIL-STD CHANNEL

1	Test Design Identifier	TD-DV1553 –040
2	Test Items	1553 Drivers
3	Input Specifications	[AD11] SRD-3.3.2.0-000
		[AD11] SRD-3.3.2.0-010
		[AD11] SRD-3.3.3.1-000
		[AD11] SRD-3.3.3.1-010
		[AD11] SRD-3.3.3.1-020
		[AD11] SRD-3.3.3.1-030
		[AD11] SRD-3.3.3.1-040
		[AD11] SRD-3.3.3.1-050
		[AD11] SRD-3.3.3.1-060
		[AD11] SRD-3.3.3.2-000
		[AD11] SRD-3.3.3.2-010
		[AD11] SRD-3.3.3.2-020
		[AD11] SRD-3.3.3.2-030
		[AD11] SRD-3.3.3.2-040
		[AD11] SRD-3.3.3.2-050
		[AD11] SRD-3.3.3.2-060
		[AD11] SRD-3.3.3.3-000
		[AD11] SRD-3.3.3.3-010
		[AD11] SRD-3.3.3.3-020
		[AD11] SRD-3.3.3.3-030
		[AD11] SRD-3.3.3.3-040
		[AD11] SRD-3.3.3.3-050
		[AD11] SRD-3.3.3.3-060
		[AD11] SRD-3.3.3.3-070
		[AD11] SRD-3.3.3.3-080
		[AD11] SRD-3.3.3.3-090
		[AD11] SRD-3.3.3.3-100
		[AD11] SRD-3.3.3.3-110
		[AD11] SRD-3.3.3.3-120
		[AD11] SRD-3.3.3.3-130
		[AD11] SRD-3.3.3.3-140
		[AD11] SRD-3.3.3.3-150
		[AD11] SRD-3.3.3.3-160
4	Output Specifications	[AD7] Event reporting section.
5	Environmental needs	PC with Condor MIL-STD-1553B PCI board and Bus tool S/W and ADSP emulator.
6	Special Procurement requirement	The correctness of the operation is given by the analysis of MILBUS message and DDC chip lookup table that has to be done by AIV people.
7	Interface Dependencies	

6.4   EEPROM DRIVER TEST CASE SPECIFICATION.

6.4.1   DELETE EEPROM SEGMENT

1	Test Design Identifier	TD-EEPROM –010
2	Test Items	EEPROM Driver.
3	Input Specifications	[AD11] SRD-3.5.0.0-000 [AD11] SRD-3.5.0.0-010
4	Output Specifications	
5	Environmental needs	PC with ADSP emulator
6	Special Procurement requirement	Correctness of the operation is given by the analysis of the generated messages that to be done by AIV people by analysis of the EEPROM content
7	Interface Dependencies	

6.4.2   WRITE EEPROM CELL

1	Test Design Identifier	TD-EEPROM –020
2	Test Items	EEPROM Driver.
3	Input Specifications	[AD11] SRD-3.5.0.0-020
4	Output Specifications	
5	Environmental needs	PC with ADSP emulator
6	Special Procurement requirement	Correctness of the operation is given by the analysis of the generated messages that to be done by AIV people by analysis of the EEPROM content
7	Interface Dependencies	

6.4.3   DELETE EEPROM CELL

1	Test Design Identifier	TD-EEPROM –030
2	Test Items	EEPROM Driver.
3	Input Specifications	[AD11] SRD-3.5.0.0-030
4	Output Specifications	
5	Environmental needs	PC with ADSP emulator
6	Special Procurement requirement	Correctness of the operation is given by the analysis of the generated messages that to be done by AIV people by analysis of the EEPROM content
7	Interface Dependencies	

6.4.4   WRITE EEPROM SEGMENT

1	Test Design Identifier	TD-EEPROM –040
2	Test Items	EEPROM Driver.
3	Input Specifications	[AD11] SRD-3.5.0.0-040
4	Output Specifications	
5	Environmental needs	PC with ADSP emulator
6	Special Procurement requirement	Correctness of the operation is given by the analysis of the generated messages that to be done by AIV people by analysis of the EEPROM content
7	Interface Dependencies	

6.4.5   ENABLE/DISABLE EEPROM PROTECTION

1	Test Design Identifier	TD-EEPROM –050
2	Test Items	EEPROM Driver.
3	Input Specifications	[AD11] SRD-3.5.0.0-000 [AD11] SRD-3.5.0.0-010 [AD11] SRD-3.5.0.0-020 [AD11] SRD-3.5.0.0-030 [AD11] SRD-3.5.0.0-040 [AD11] SRD-3.6.0.0-000
4	Output Specifications	
5	Environmental needs	PC with ADSP emulator
6	Special Procurement requirement	Correctness of the operation is given by the analysis of the generated messages that to be done by AIV people by analysis of the EEPROM content
7	Interface Dependencies	

6.4.6   COPY PROGRAM IN EEPROM

1	Test Design Identifier	TD-EEPROM –060
2	Test Items	EEPROM Driver.
3	Input Specifications	[AD11] SRD-3.5.0.0-000 [AD11] SRD-3.5.0.0-010 [AD11] SRD-3.5.0.0-020 [AD11] SRD-3.5.0.0-030 [AD11] SRD-3.5.0.0-040 [AD11] SRD-3.6.0.0-000
4	Output Specifications	
5	Environmental needs	PC with ADSP emulator
6	Special Procurement requirement	Correctness of the operation is given by the analysis of the generated messages that to be done by AIV people by analysis of the EEPROM content
7	Interface Dependencies	

6.5   WATCHDOG DRIVER TEST CASE.

6.5.1   DELAY SETTING TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DRVWD –010
2	Test Items	Watchdog Delay Setting.
3	Input Specifications	[AD11] SRD-3.6.0.0-000 [AD11] SRD-6.4.0.0-000
4	Output Specifications	
5	Environmental needs	Test Equipment CPU board
6	Special Procurement requirement	Correctness of the operation is given by the analysis of the generated messages that to be done by AIV people
7	Interface Dependencies	

6.5.2   DELAY PROGRAMMING TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DRVWD –020
2	Test Items	Watchdog Delay Programming.
3	Input Specifications	[AD11] SRD-3.6.0.0-010 [AD11] SRD-6.4.0.0-000
4	Output Specifications	
5	Environmental needs	Test Equipment CPU board
6	Special Procurement requirement	Correctness of the operation is given by the analysis of the memory map and generated messages that to be done by AIV people
7	Interface Dependencies	

6.5.3   WATCHDOG REFRESH TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DRVWD –030
2	Test Items	Watchdog Refresh.
3	Input Specifications	[AD11] SRD-3.6.0.0-020 [AD11] SRD-6.4.0.0-000
4	Output Specifications	
5	Environmental needs	Test Equipment CPU board
6	Special Procurement requirement	Correctness of the operation is given by the analysis of the generated messages that to be done by AIV people
7	Interface Dependencies	

6.5.4   WATCHDOG STATUS TEST CASE SPECIFICATION.

1	Test Design Identifier	TD-DRVWD –040
2	Test Items	Watchdog Status.
3	Input Specifications	[AD11] SRD-3.6.0.0-000 [AD11] SRD-6.4.0.0-000
4	Output Specifications	
5	Environmental needs	Test Equipment CPU board
6	Special Procurement requirement	Correctness of the operation is given by the analysis of the

		memory map and generated messages that to be done by AIV people
7	Interface Dependencies	

7. TEST PROCEDURES.

7.1   DPU BOOT SFWB TEST PROCEDURES.

7.1.1   BOOT SW TEST PROCEDURE.

Step	Test Procedure Identifier	TD-BOOTSW –010
	Purpose	Verify the uploading of an Application Program from EEPROM.
	Special Requirement	For this operation is requested an Open Box condition to connect the DSP emulator.
Procedures Steps	
1		Program the EEPROM mezzanine using the SW Test drivers. Dump the EEPROM contained from 0x80000000 to 0x800017ff. Insert a programmed EPROM (FIRST DPU BOOT SW ver 2.0.) on the HSO/FIRST DPU board. Power On. Connect the ADSP Emulator Reset the Board. Open the Program Window. Verify that the BOOTSW program has been uploaded in the first 1555 cells in program memory.
2		Perform a chip reset by emulator Start program Set breakpoint before of jumping to "Application Program" Verify that the program has been uploaded dumping the PM area from 0 to 0x100 cells and from 0x4000 to 0x4acc cell and comparing it with the instruction inside the EXE file.
3		Set a new breakpoint before of reprogramming the EEPROM Start the program Verify that the EEPROM has been cleaned and the first six pages are blanks.
4		Start the program Wait until the end of program Verify that the program has been correctly performed comparing the new programmed EEPROM with the values previously dumped

		N° Doc:	DPU-PL-CGS-002
	HSO/FIRST-DPU	*Doc N°:*	
CARLO GAVAZZI		Ediz.: **1**	Data: **5/04/2002**
		Issue:	*Date:*
CARLO GAVAZZI SPACE SpA	**DPU-SW VERIFICATION AND VALIDATION PLAN/ACCEPTANCE TEST**	Pagina **38**	di **61**
		Page	*of*

7.1.2 DM TEST PROCEDURE.

Step	Test Procedure Identifier	TD-BOOTSW –020
	Purpose	Verify DM errors detection during initialization
	Special Requirement	For this operation is requested an Open Box condition to connect the DSP emulator.
	Procedure steps	
1	Insert a programmed EPROM (FIRST DPU BOOT SW ver 2.0.) on the HSO/FIRST DPU board. Power On. Connect the ADSP Emulator Reset the Board. Open the Program Window and Data Window Verify that the BOOTSW program has been uploaded in the first 1555 cells in program memory.	
2	Perform a chip reset by emulator Start program Set breakpoint before of reading the data from Data memory Write the 0xFFFFFFFF data in the DM cell Start program Verify that the Boot SW detects the DM error verifying that a TM error message is sent to the FIRST TE containing the incorrect DM page address.	
3	Verify that the BOOT SW is waiting a TC from FIRST TE checking that the "Application Program" has not been also uploaded (from 0x4000 to 0x4acc the PM contains spurious data)	

7.1.3 PM TEST PROCEDURES

Step	Test Procedure Identifier	TD-BOOTSW –025
	Purpose	Verify possible PM error detection during initialization
	Special Requirement	For this operation is requested an Open Box condition to connect the DSP emulator.
	Procedure steps	
1	Insert a programmed EPROM (FIRST DPU BOOT SW ver 2.0.) on the HSO/FIRST DPU board. Detach the Emulator and reconnect in order to erase previous value in PM. Power On. Connect the ADSP Emulator Reset the Board. Open the Program Window and Data Window Verify that the BOOTSW program has been uploaded in the first 1555 cells in program memory.	
2	Perform a chip reset by emulator Start program Set breakpoint before of reading the data from Program memory Write the 0xFFFFFFFFFFFF data in the PM cell Start program Verify that the Boot SW detects the PM error verifying that a TM error message is sent to the FIRST TE containing the incorrect PM page address.	
3	Verify that the BOOT SW is waiting a TC from FIRST TE checking that the "Application Program"	

	has not been also uploaded (from 0x4000 to 0x4acc the PM contains spurious data)

7.1.4 EEPROM TEST PROCEDURE.

Step	Test Procedure Identifier	TD-BOOTSW –030
	Purpose	Verify that the "Application Software" is uploaded when an error in the first partition is found.
	Special Requirement	For this operation is requested an Open Box condition to connect the DSP emulator.
		Procedure steps
1	Program the EEPROM mezzanine using the SW Test drivers. Dump the EEPROM contained from 0x80000000 to 0x800017ff. Insert a programmed EPROM (FIRST DPU BOOT SW ver 2.0.) on the HSO/FIRST DPU board. Detach the Emulator and reconnect in order to erase previous value in PM. Power On. Connect the ADSP Emulator Reset the Board. Open the Program Window and Data Window ⏎ Verify that the BOOTSW program has been uploaded in the first 1555 cells in program memory.	
2	Perform a chip reset by emulator Read the value at the EEPROM address 0x80000020 Write a different value (i.e 0xFFFFAAAA) at the EEPROM address 0x80000020 Set a break point before of jumping to the "Application Program" execution. Start program ⏎ Verify that the program has been uploaded, dumping the PM area from 0 to 0x100 cells and from 0x4000 to 0x4acc cell and comparing it with the instruction inside the EXE file.	
3	Set a new breakpoint before of reprogramming the EEPROM Start the program ⏎ Verify that the EEPROM has been cleaned and the first six pages are blanks.	
4	Start the program Wait until the end of program ⏎ Verify that the program has been correctly performed comparing the new programmed EEPROM with the values previously dumped Verify that the values 0xFFFFAAAA at the 0x80000020 is equal to the value read at step 2	
5	Restart the system (power off) Insert a programmed EPROM (FIRST DPU BOOT SW ver 2.0.) on the HSO/FIRST DPU board. Detach the Emulator and reconnect in order to erase previous value in PM. Power On. Connect the ADSP Emulator Reset the Board. Open the Program Window and Data Window ⏎ Verify that the BOOTSW program has been uploaded in the first 1555 cells in program memory.	
6	Perform a chip reset by emulator Read the value at the EEPROM address 0x8003fc20 Write a different value (i.e 0xFFFFAAAA) at the EEPROM address 0x8003fc20 Set a break point before of jumping to the "Application Program" execution. Start program	

	Verify that the program has been uploaded, dumping the PM area from 0 to 0x100 cells and from 0x4000 to 0x4acc cell and comparing it with the instruction inside the EXE file.
7	Set a new breakpoint before of reprogramming the EEPROM Start the program Verify that the EEPROM has been cleaned and the first six pages are blanks.

8	Start the program Wait until the end of program Verify that the program has been correctly performed comparing the new programmed EEPROM with the values previously dumped Verify that the values 0xFFFFAAAA at the 0x8003fc20 is equal to the value read at step 5
9	Restart the test procedure (Power off) Detach the Emulator and reconnect in order to erase previous value in PM. Power On. Connect the ADSP Emulator Reset the Board. Open the Program Window and Data Window Verify that the BOOTSW program has been uploaded in the first 1555 cells in program memory.
10	Perform a chip reset by emulator Read the value at the EEPROM address 0x80000060 and 0x8003fcAA Write a different value (i.e 0xFFFFAAAA) at the EEPROM address 0x80000060 and 0x8003fcAA Set a break point before of jumping to the "Application Program" execution. Start program Verify that the program has not been uploaded, and the "Application Program" has not been copied in the 0x4000 to 0x4acc cells. Verify that a TM error message is generated indicating the EEPROM page address failed And the Checksum expected and computed..

7.1.5   EEPROM TO PM LOAD TEST PROCEDURE.

Step	Test Procedure Identifier	TD-BOOTSW –040
	Purpose	Verify an error is detected during the uploading of an Application Program from EEPROM to PM.
	Special Requirement	For this operation is requested an Open Box condition to connect the DSP emulator.
	Procedure steps	
1	Program the EEPROM mezzanine using the SW Test drivers Insert a programmed EPROM (FIRST DPU BOOT SW ver 2.0.) on the HSO/FIRST DPU board. Power On. Connect the ADSP Emulator Reset the Board. Open the Program Window. Verify that the BOOTSW program has been uploaded in the first 1555 cells in program memory.	
2	Perform a chip reset by emulator Set breakpoint before of starting the EEPROM to PM upload Start program No checks is requested	
3	Write at the EEPROM address 0x80000820 the value 0xFFFFFFFF Stat program Stop program and Verify that a TM error message is sent by means of MIL1553 to FIRST TE indicating the failed PM page upload. Verify that the "Application Program" has been partially uploaded: page 1,2 are ok page 3 is failed checking the PM content.	

7.1.6 DM TO PM LOAD TEST PROCEDURE.

Step	Test Procedure Identifier	TD-BOOTSW –050
	Purpose	Verify an error is detected during the uploading of an Application Program from DM to PM.
	Special Requirement	For this operation is requested an Open Box condition to connect the DSP emulator.
	Procedure steps	
1	Reset the EEPROM content Generate using Tcgen program the DM pages that will be uploaded Insert a programmed EPROM (FIRST DPU BOOT SW ver 2.0.) on the HSO/FIRST DPU board. Power On. Connect the ADSP Emulator Reset the Board. Open the Program Window and Data Window. Verify that the BOOTSW program has been uploaded in the first 1555 cells in program memory.	
2	Perform a chip reset by emulator Upload the DM pages in Data Memory starting from 0x00004000 Set breakpoint before of the DM test Start program No check is required	
3	Change the value in the memory cell in order to fail the DM test Start the program Verify that a TM message is sent to FIRST TE	
4	Send a TC in order to force the DM to PM program copy Set a breakpoint before of jumping to "Application Program" execution Verify that the "Application Program" is correctly copied at the following address: Pm 0x0 to 0x100 Pm 0x4000 to 0x4acc Dumping the PM content and comparing it with the Application program exe file	
5	Start Program Verify that the EEPROM has been reprogrammed by means of the "Application Program" uploaded	

7.2   1355 DRIVER TEST PROCEDURE.

7.2.1   1355 POWER ON TEST PROCEDURE.

1	Test Procedure Identifier	TD-DRV1355 –010
2	Purpose	Verify the correct functioning of 1355 Power On.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Power-On.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Power-On-log.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet. For several information about the other packet's word see Annex A2.	

7.2.2   1355 RESET LINK TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –020
2	Purpose	Verify the possibility to reset the SMCS332 link.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
	Procedure steps	
4	Procedure steps Power On. Load on Bus Tools Software the file Reset-link.btd Run on Bus Tools the network Wait to receive the last packet from remote terminal Off-line verification: Check in the log file Reset-link.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.3   1355 TIME-OUT TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –030
2	Purpose	Verify the possibility to set time-out value for the SMCS332.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On Load on Bus Tools Software the file Time-out.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Time-out.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.4   1355 OPEN LINK TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355–040
2	Purpose	Verify the possibility to open each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On Load on Bus Tools Software the file Open-link.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Open-link.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet	

7.2.5   1355 CLOSE LINK TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355–050
2	Purpose	Verify the possibility to close each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.

4	Power On
	Load on Bus Tools Software the file Close-link.btd.
	Run on Bus Tools the network.
	Wait to receive the last packet from remote terminal
	Off-line verification:
	Check in the log file Close-link.bmd the following:
	1) at what function the packet answer are shown in ID field.
	2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C.
	3) the packet size is shown in third word of packet.

7.2.6 1355 START LINK AS MASTER TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355–060
2	Purpose	Verify the possibility to start, as master, each of the three SMCS links using different speed values.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On	
	Load on Bus Tools Software the file Start-Link-as Master.btd.	
	Run on Bus Tools the network.	
	Wait to receive the last packet from remote terminal.	
	Off-line verification:	
	Check in the log file Start-Link-as Master.bmd the following:	
	1) at what function the packet answer are shown in ID field.	
	2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C.	
	3) the packet size is shown in third word of packet.	

7.2.7 1355 START LINK AS SLAVE TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –070
2	Purpose	Verify the possibility to start, as slave, each of the three SMCS links using different speed values.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On.	
	Load on Bus Tools Software the file Start-Link-as-slave.btd.	
	Run on Bus Tools the network.	
	Wait to receive the last packet from remote terminal	
	Off-line verification:	
	Check in the log file Start-Link-as-slave.bmd the following:	
	1) at what function the packet answer are shown in ID field.	
	2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C.	
	3) the packet size is shown in third word of packet.	

7.2.8 1355 STOP LINK TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –080

2	Purpose	Verify the possibility to start, as slave, each of the three SMCS links using different speed values.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Start-Link-as-slave.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal Off-line verification: Check in the log file Start-Link-as-slave.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet..	

7.2.9   1355 GET STATUS REGISTER TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –080
2	Purpose	Verify the possibility to get the status of register for each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On Load on Bus Tools Software the file Get-Status-Register.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal Off-line verification: Check in the log file file Get-Status-Register.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.10 1355 WRITE LINK TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355–070
2	Purpose	Verify the possibility to transmit data over each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Write-Link.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal Off-line verification: Check in the log file Write-Link.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.11 1355 READ LINK TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –100
2	Purpose	Verify the possibility to receive data over each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Read-Link.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Read-Link.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.12 1355 GET LINK STATUS TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –110
2	Purpose	Verify the possibility to return the current transmit/receive link status for each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Get-Link-Status.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Get-Link-Status.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.13 1355 READ PACKET TEST PROCEDURE.

	Test Procedure Identifier	TD- DRV1355–120
2	Purpose	Verify the possibility to receive data over each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Read-packet.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Read-packet.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.14 1355 GET LAST READ SIZE TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355–130
2	Purpose	Verify the possibility to get the number of received data over each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Get-Last-Read-Size.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Get-Last-Read-Size.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.15 1355 GET LAST READ WRITE SIZE TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –130
2	Purpose	Verify the possibility to get the number of transmitted data over each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Get-Last-Write-Size.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Get-Last-Write-Size.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.16 1355 GET LAST PACKET NUMBER TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –140
2	Purpose	Verify the possibility to get the number of received packet over each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.

4	Power On.
	Load on Bus Tools Software the file Get-Last-Packet-Num.btd.
	Run on Bus Tools the network.
	Wait to receive the last packet from remote terminal.
	Off-line verification:
	Check in the log file Get-Last-Packet-Num.bmd
	the following:
	1) at what function the packet answer are shown in ID field.
	2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed.
	To find error condition see Annex C.
	3) the packet size is shown in third word of packet.

7.2.17 1355 WRITE TO BOARD MEMORY TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –160
2	Purpose	Verify the possibility to write in DPRAM for each of the three SMCS links.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Write-to-board Mem.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Write-to-board Mem.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.18 1355 READ FROM BOARD MEMORY TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355–170
2	Purpose	Verify the possibility to read from DPRAM of SMCS device.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Write-to-board Mem.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Write-to-board Mem.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.19 1355 WRITE REGISTER TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355–180
2	Purpose	Verify the possibility to write in a generic register of SMCS device.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Write-to-board Mem.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Write-to-board Mem.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.2.20 1355 READ REGISTER TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRV1355 –190
2	Purpose	Verify the possibility to read the value contained in a generic register of SMCS device.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file Read-Reg.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file Read-Reg.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.3   WATCHDOG TEST PROCEDURE.

7.3.1   DELAY SETTING TEST PROCEDURE.

1	Test Procedure Identifier	TD- DRVWD –010
2	Purpose	Verify the correct functioning of Watchdog Delay Setting.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file WD-Delay.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file WD-Delay.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.3.2   DELAY PROGRAMMING TEST PROCEDURE.

1	Test Procedure Identifier	TD-DRVWD–020
2	Purpose	Verify the correct functioning of Watchdog Delay Programming.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file WD-Delay-fail.btd Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file WD-Delay-fail.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.3.3   WATCHDOG REFRESH TEST PROCEDURE.

1	Test Procedure Identifier	TD-DRVWD –030
2	Purpose	Verify the correct functioning of Watchdog refresh.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On Load on Bus Tools Software the file WD-Refresh.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file WD-Refresh.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.3.4   WATCHDOG STATUS TEST PROCEDURE.

1	Test Procedure Identifier	TD-DRVWD –040
2	Purpose	Verify the correct functioning of Watchdog status.
3	Special Requirement	For this operation is requested the use of S/W Stub and Test Equipment.
4	Power On. Load on Bus Tools Software the file WD-Status.btd. Run on Bus Tools the network. Wait to receive the last packet from remote terminal. Off-line verification: Check in the log file WD-Status.bmd the following: 1) at what function the packet answer are shown in ID field. 2) the success are shown in Error Code field. If Error Code is different from zero, the function is failed. To find error condition see Annex C. 3) the packet size is shown in third word of packet.	

7.4 MIL-STD 1553 SW DRIVER TEST PROCEDURE

7.4.1 OPEN MIL-STD 1553 CHANNEL

Step	Test Procedure Identifier	TD-DRV1553 –010
	Purpose	Verify the configuration of the MIL register and look up table
	Special Requirement	.
		Procedure steps
1		Upload the MIL-STD-1553 SW test using the ADSP emulator Connect the MIL1553 between board and Test Equipment and start the CONDOR BUS TOOL Program Set Remote terminal = 7 Set breakpoint after the Open procedure Start the program Verify that the Mil registers are configured as defined in the specification The DDC chip is configured as Remote Terminal
2		Set a new breakpoint after the mil configuration procedure Verify that the Mil registers are configured as defined in the specification The Look Up Table as the following configuration - The SA 10, 11-14, 27 Rx message has been configured pointing to Mil1553 memory free - The SA 10,11-26,27 Tx message has been configured pointing to Mil1553 memory free - The related SACW has been configured the message are configured in buffer message mode/or single buffer mode Verify that the pointed memory space are not overlapped but they are correctly allocated Verify that the Interrupt on Broadcast synchronize message mode code with data is enabled Verify that the memory configuration of the MIL follows the Mil DDC specification.
3		Set breakpoint after the Rtrun procedure Verify that the Mil Stack memory is cleared 0x8F000000 0x80000100

7.4.2   TX MIL-STD 1553 MESSAGE

Step	Test Procedure Identifier	TD-DRV1553 –020
	Purpose	Verify the Tx messages transmission
	Special Requirement	.
	Procedure steps	
1	Upload the MIL-STD-1553 SW test using the ADSP emulator Connect the MIL1553 between board and Test Equipment and start the CONDOR BUS TOOL Program Set Remote terminal = 7 Set breakpoint after the Open procedure Start the program Set the CONDOR BUS TOOL Tx messages in order to acquire the SA 10, 11-26, 27 message Verify that the messages are correctly received (no random value are acquired) Verify that the messages in the SA 11- 26 are composed of 32 words Verify that the messages in the SA 10, 27 are composed of 2 words	
2	Acquire Tx messages by means of the CONDOR BUS TOOLS Verify that the messages are correctly received (the value are incremented with respect to the Previous acquisition) Verify that the messages in the SA 11- 26 are composed of 32 words Verify that the messages in the SA 10, 27 are composed of 2 words Verify stopping the emulator that MIL1553 Stack memory contains the block message words	

7.4.3   RX MIL-STD 1553 MESSAGE

Step	Test Procedure Identifier	TD-DRV1553 –030
	Purpose	Verify the Rx messages reception
	Special Requirement	.
	Procedure steps	
1	Upload the MIL-STD-1553 SW test using the ADSP emulator Connect the MIL1553 between board and Test Equipment and start the CONDOR BUS TOOL Program Set Remote terminal = 7 Set breakpoint after the Open procedure Start the program Set the CONDOR BUS TOOL Rx messages in order to acquire the SA 10, 11-24, 27 message Verify that the messages are correctly transmitted to the ADSP Board checking the MIL DDC memory Verify that the messages in the SA 11- 26 are composed of 32 words Verify that the messages in the SA 10, 27 are composed of 2 words	
2	Transmit Rx messages by means of the CONDOR BUS TOOLS Verify that the messages are correctly transmitted to the ADSP Board checking the MIL DDC memory Verify that the messages in the SA 11- 26 are composed of 32 words Verify that the messages in the SA 10, 27 are composed of 2 words	

	Verify stopping the emulator that MIL1553 Stack memory contains the block message words

7.4.4 CLOSE MIL-STD CHANNEL

Step	Test Procedure Identifier	TD-DRV1553 –040
	Purpose	Verify the Rx messages reception
	Special Requirement	.
		Procedure steps
1	Upload the MIL-STD-1553 SW test using the ADSP emulator Connect the MIL1553 between board and Test Equipment and start the CONDOR BUS TOOL Program Set Remote terminal = 7 Set breakpoint after the Open procedure Start the program Send and Acquire Rx and Tx MIL messages Verify that all the messages are received - SA Rx 10,11-14,27 - SA Tx 10,11,26,27	
2	Exit the program Verify that the Mil register are reset	

7.5 EEPROM DRIVER TEST PROCEDURE

7.5.1 DELETE EEPROM SEGMENT

Step	Test Procedure Identifier	TD-DRVEPRM –010
	Purpose	Deleting of EEPROM segment
	Special Requirement	.
	Procedure steps	
1	Upload the EEPROM test program using the ADSP emulator Set a breakpoint after a Delete Segment function call Write 10 times the 0xFFFFFFFF value beginning from the 0x80000000 address Start the program Verify (when the breakpoint is stricken) that the segment beginning at 0x80000000 until 0x800003FF address is cleaned	

7.5.2 WRITE EEPROM CELL

Step	Test Procedure Identifier	TD-DRVEPRM –020
	Purpose	Write EEPROM single cell inside the segment
	Special Requirement	.
	Procedure steps	
1	Upload the EEPROM test program using the ADSP emulator Set a breakpoint after a write segment cell function call Start the program Verify (when the breakpoint is stricken) that the segment beginning at 0x80000C00 until 0x80000FFF address is written register the frame check sequence at the 0x80000C06 address and the value at 0x8000C17	
2	Set a breakpoint after the "write eeprom cell" function call Start program Verify that the 0xFFFFFFFF value is written at 0x80000C17 address and the DM Frame Check Sequence and PM Frame Check Sequence are changed with respect to the previous value	
3	Set a Breakpoint after the "write eeprom cell" function call The original value of the segment is written at 0x80000c17 address Start program Verify that the 0x00000010 value is written at 0x80000C17 Verify the DM Frame Check sequence and PM Frame sequence are the same of the step 1	
4	Set a breakpoint before the "write eeprom cell" function call which writes the value 0xFFFFFFFF at the 0x80000C06 Read the value in 0x80000c06 Start Program No check is required	
5	Set breakpoint after the "write eeprom cell" function call	

	Verify that the 0xFFFFFFFF is written at 0x80000c06 address; Verify that the DM frame check sequence is changed with respect to value reported in step 4 Verify that the PM check sequence is equal to the value reported in Step 4

6	Write the original value at the 0x80000C06 Verify that DM FCS is equal to value reported in step 4 Verify that PM FCS is equal to the value reported in step 4

7.5.3 DELETE EEPROM CELL

Step	Test Procedure Identifier	TD-DRVEPRM –020
	Purpose	Deleting of EEPROM single cell inside a segment
	Special Requirement	.
		Procedure steps
1		Upload the EEPROM test program using the ADSP emulator Set a breakpoint after a write segment cell function call Start the program Verify that the segment beginning at 0x80000C00 until 0x80000FFF address is written register the frame check sequence at the 0x80000C06 address and the value at 0x8000C17
2		Set a breakpoint after the "delete eeprom cell" function call Start program Verify that the 0x000000000 value is written at 0x80000C17 address and the DM Frame Check Sequence and PM Frame Check Sequence are changed with respect to the previous value
3		Set a Breakpoint after the "write eeprom cell" function call The original value of the segment is written at 0x80000c17 address Start program Verify that the 0x00000010 value is written at 0x80000C17 Verify the DM Frame Check sequence and PM Frame sequence are the same of the step 1

4	Set a breakpoint before the "delete eeprom cell" function call which writes the value 0x00000000 at the 0x80000C06 Read the value in 0x80000c06 Start Program No check is required
5	Set breakpoint after the "delete eeprom cell" function call Verify that the 0x00000000 is written at 0x80000c06 address; Verify that the DM frame check sequence is changed with respect to value reported in step 4 Verify that the PM check sequence is equal to the value reported in Step 4
6	Write the original value at the 0x80000C06 Verify that DM FCS is equal to value reported in step 4 Verify that PM FCS is equal to the value reported in step 4

7.5.4 WRITE EEPROM SEGMENT

Step	Test Procedure Identifier	TD-DRVEPRM –030
	Purpose	Verify the EEPROM segment write
	Special Requirement	.
	Procedure steps	
1	Upload the EEPROM test program using the ADSP emulator Set a breakpoint after a Delete Segment function call Start the program Verify that the segment beginning at 0x80000C00 until 0x80000FFF address is cleaned	
2	Set breakpoint after "write segment function" Start Program Verify that a EEPROM segment is written at 0x80000C00 until xù0x80000FFF Verify that value of the segment are incremented by one starting from 0x8000C07 Verify that the first six word are the same defined inside the header	

3	Verify that in the header are included the field as follow
	   - current page
	   - Total Page
	   - NextPage in EEPROM
	   - Application S/W start address
	   - Application S/W Frame Check Sequence
	   - PM start address
	   - PM length
	   - DM FCS
	   - PM FCS
	   - Boot Option
	Verify that the word at 0x80000C05 address is the PM FCS (MSB 16 bits) and DM FCS (LSB 16 bits)	

7.5.5 ENABLE/DISABLE EEPROM PROTECTION

Step	Test Procedure Identifier	TD-DRVEPRM –040
	Purpose	Verify Enable/Disable EEPROM protection
	Special Requirement	.
	Procedure steps	
1	Upload the EEPROM test program using the ADSP emulator Set a breakpoint after the "Enable Protection EEPROM" function call Verify that is possible to write a random data in EEPROM using the emulator Start the program Verify that is no more possible to write a data in EEPROM (EEPROM is protected)	
2	Set breakpoint after the "Disable Protection" function call Verify that is possible to write in EEPROM the data	

7.5.6   COPY PROGRAM IN EEPROM

Step	Test Procedure Identifier	TD-DRVEPRM –050
	Purpose	Uploading in EEPROM of a generic program in Program Memory
	Special Requirement	.
		Procedure steps
1		Upload the EEPROM test program using the ADSP emulator Start the program Wait the end of program
2		Verify that the first EEPROM page is written starting from 0x80000000 address Verify that the header has the following values - PM address = 0000 - PM length = 0x100 - Current Eeprom page = 1 - Total Eeprom pages = 6 - Next Eeprom page = 0x80000400 Verify that in the segment are recorded the first 256 instructions (interrupt table) (Primary boot)
3		Verify that the Last EEPROM page is written starting from 0x8003FC00 address and the content is the same of the page in step 2 Verify that the header has the following values - PM address = 0000 - PM length = 0x100 - Current Eeprom page = 1 - Total Eeprom pages = 6 - Next Eeprom page = 0x80000400 Verify that in the segment are recorded the first 256 instructions (interrupt table) (secondary Boot)
4		Verify that the EEPROM page is written starting from 0x8003FC00 address Verify that the header has the following values - PM address = 4000 - PM length = 0x2A6 - Current Eeprom page = 2 - Total Eeprom pages = 6 - Next Eeprom page = 0x80000800 Verify that in the segment are recorded 0x2A6 instruction starting from PM address 0x4000
5		Verify that the EEPROM page is written starting from 0x8003FC00 address Verify that the header has the following values - PM address = 42A6 - PM length = 0x2A6 - Current Eeprom page = 3 - Total Eeprom pages = 6 - Next Eeprom page = 0x80000C00 Verify that in the segment are recorded 0x2A6 instruction starting from PM address 0x42A6
6		Verify that the EEPROM page is written starting from 0x8003FC00 address Verify that the header has the following values - PM address = 454C - PM length = 0x2A6 - Current Eeprom page = 4 - Total Eeprom pages = 6 - Next Eeprom page = 0x80001000

	Verify that in the segment are recorded 0x2A6 instruction starting from PM address 0x454C

7	Verify that the EEPROM page is written starting from 0x8003FC00 address
	Verify that the header has the following values
	    - PM address = 47F2
	    - PM length = 0x2A6
	    - Current Eeprom page = 5
	    - Total Eeprom pages = 6
	    - Next Eeprom page = 0x80000C00
	Verify that in the segment are recorded 0x2A6 instruction starting from PM address 0x47F2