EPM
Standard Protocols
Definition

for the
European Physiology Modules Facility
(EPM)

Doc. No.:	**EPM-OHB-SP-0005**		
Issue:	**4**	**Date:**	**30.04.2010**
Revision:	**-**	**Date:**	**-**

Prepared by: D. Roßk .... **Date:** 30.04.2010
D. Rosskamp/ OHB System AG

Checked by: O. Amend .... **Date:** 30.04.2010
O. Amend / OHB System AG

Product Assurance: .... **Date:** 30.04.2010
J. Mathes / OHB System AG

Project Management: M. Berg .... **Date:** 30.04.2010
M. Berg / OHB System AG

Approved by IOT: .... **Date:** ....
- / IOT

Approved by ESA: .... **Date:** ....
- / ESA

Copying of this document, and giving it to others and the use or communication of the contents, thereof, are forbidden without express authority. Offenders are liable to the payment of damages. All rights are reserved in the event of the grant of a patent or the registration of utility model or design.	**OHB-System AG** D-28359 Bremen Universitätsallee 27-29	Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts ist nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte für den Fall der Patenterteilung oder Gebrauchsmuster-Eintragung vorbehalten.

© 2010 OHB System AG

Distribution List

Name	*Company / Organisation*
DCC	OHB
Local ECB	OHB

© 2010 OHB System AG

Document Change Record

Issue/Rev.	*Date*	*Affected Section / Para*	*Change Reason / Description*
Draft	29/03/01	ALL	Initial Issue
1/-	08/06/01	ALL	Modbus Protocol introduced, USB Interface introduced
1/A	17/10/01	Section 5.3, 5.4, 5.5	Modbus Protocol Layer Extension for function code 0x41
1/B	31/01/02	Section 3	FCC LAN connection introduced in Figure 3.1
		Section 4.1	IP-Addresses defined
		Section 4.2	FCC introduced in LAN concept, Software unit identification defined
		Section 4.3.3	Remote shutdown packet type deleted
		Section 5	RS-485 settings defined
		Section 5.4	Transfer frame for broadcast messages introduced, Time Set and ancillary data transfer are transferred via broadcast, Use of “Type Of Data”-Field clarified
		Section 5.4.10	Application data references and description added
		Section 5.5	Use of “Previous application data package acknowledge” clarified Structure for application data type “Messages contains a 32 bit unsigned integer count for the number of messages that are transferred
		Section 5.5.10	Application data references and description added
2/-	05/06/02	Section 1.1	Scope extended to Laptop internal communication and USB communication
		Section 2.1	EPM-OHB-MAN-004 introduced in AD list
		Section 3	Document restructured: Section 3 describes all EPM standards that have been documented in EPM-OHB-LI-0039 before, definition of “kilo” and “Mega” for bit and byte related units
		Section 3.11	Not used deleted from Command Priority table
		Section 3.13	Not used deleted from Telemetry Destination table
		Section 3.14	Not used deleted from Telemetry Processing table

© 2010 OHB System AG

Issue/Rev.	*Date*	*Affected Section / Para*	*Change Reason / Description*
		Section 3.15	Table Version Coding: Nibble Numbering starts from 0
		Section 4.3	SW Unit ID is coded in one octet
		Section 4.3.3	SM-MMI Control Specific types introduced, Time interval for ALIVE Packets specified
		Section 4.3.6	Coarse time removed from ALIVE packet, reference to AD list for application data added
		Section 4.4	SNTP server is running on SMSC
		Section 5.2.7	Section deleted in favour of Section 3.9
		Section 5.2.8	Data and Control Functions are not supported
		Section 5.2.9	Diagnostics Subfunctions are not supported
		Section 5.2.10	Exception Responses are not supported
		Section 5.3	Application data renamed to User Data in order avoid misunderstanding
		Section 5.4.8	Note on treatment of RS-485 unique TC Counter added
		Section 5.4.9.1	Ancillary data details and link for TC structure added.
		Section 5.5.5	Calculation of counter detailed
		Section 5.5.6	Calculation of counter detailed
		Section 5.5.10	Alive packets contribute to the data rate
		Section 5.5.11	Unit for HRD data rate is kBit/s
		Section 5.14.1.1	Note on time packet generation added
		Section 5.14.1.3	Link for message and TM structure introduced
		Section 6	USB Transfer Frame introduced
		Section 7	Section on Application Data Types added. This is a copy from AD 3. Changes are only highlighted if a change in AD3 is introduced
		Section 7.1.4	Detail on check of time-tag flags provided
		Section 7.1.10	Maximum number of commands per sequence is 100
		Section 7.1.16	Coarse time is GPS time
		Section 7.2.11	Coarse time is GPS time
		Section 7.2.16	Details on TM Processing field provided
		Section 7.3.7	Coarse time is GPS time

© 2010 OHB System AG

Issue/Rev.	***Date***	***Affected Section / Para***	***Change Reason / Description***
		Section 8	SM-MMI Control Data Types introduced. The section is completely new.
3/-	15/11/2002	Section 1.1	File transfer is within the scope of this document
		Section 1.2	Figure 1.1 includes TM stream of Files, Interface Transparency concept omitted
		Section 3.3	Details on application data identifier coding added
		Section 3.4	Format instruction does not support octets, strings and byte arrays contain intrinsic length information and length check option is introduced, double are supported in TC, clarification on float format, check instruction definition added
		Section 3.6	Subsystem ID for VU added, Position Information added to SAC number
		Section 3.7	Device inactive and Unknown Mode added, conversion for Columbus MilBus Data added
		Section 3.10	Columbus MTL removed as possible command source
		Section 3.13	Telemetry Destination: INTERNAL introduced
		Section 3.14	Telemetry processing is automatically handled by SMSC, control field is deleted accordingly
		Section 3.16	Multiple check bit replaced by valid bit
		Section 4	TCP/IP protocol for USB introduced
		Section 4.1	Network part for USB and host part for VU introduced
		Section 4.2	Details on LTU internal TM distribution, Software Unit ID's for USB introduced
		Section 4.3.2	Inserted ID is ID of sender
		Section 4.3.5	Maximum length of EPM TCP/IP frame is 1412 octets
		Section 5.1	RS-485 Address of VU added
		Section 5.3	Requirements on Content of User Data clarified
		Section 5.4.5	Contact status for each band is provided
		Section 5.4.9.1	Ancillary data are described in section 7.4

© 2010 OHB System AG

Issue/Rev.	*Date*	*Affected Section / Para*	*Change Reason / Description*
		Section 5.5	Ancillary/HK Request Status deleted, covered by destination field of TM and broadcast ancillary data distribution, SM Critical HK-Data deleted, details on HK-data distribution in section 5.5.14.1.3
		Section 6	TCP/IP protocol is used for USB
		Section 7.2	TM Processing deleted, Receiver Subsystem ID and Unit ID added
		Section 7.4	Ancillary Data Description and accuracy of ELCS data
		Section 8	Software Unit ID added to directive header, more details on relation to IAP instructions, timeout for directive acknowledge defined, SET MMI Visibility introduced
		Section 9	EPM File Transfer added
3/A	20/12/2004	Section 2.1	Add AD6 CPI-RIBRE-ICD-0003-PIRN-0004
		Section 4.1	Add Port number
		Section 4.3.3	Add packet type for HRD Bitstream
		Section 7.4	Modify Ancillary Data Description according CPI-RIBRE-ICD-0003-PIRN-0004
3/B	29/04/2008	Section 7.4.3	‘Partial CO2 Pressure In Cabin’ changed according CPI-RIBRE-ICD-0003-PIRN-0004, 1/A, 26.08.2004
		Section 9.1	Max. file size for EPM file transfer added
		Distribution List	List updated for Phase E
4/-	30/04/2010	General	Pagination and font standardized
		Section 3.6	Clarification for XSMs
		Section 4.1	Clarification for XSMs
		Section 4.1	Clarification for XSMs
		Table 4-1	Startup IP address for MEEMM and Cardiolab
		Section 4.2	Startup IP address for MEEMM and Cardiolab
		Section 4.2	Clarification for XSMs
		Section 5.1	Clarification for XSMs

© 2010 OHB System AG

Issue/Rev.	*Date*	*Affected Section / Para*	*Change Reason / Description*
		Section 7.2.17	reference to section 3.9 added for clarification
		Section 7.2.18	Typing error corrected
		Section 7.2.19	Typing error corrected
		Section 9.3.3.3	Table 9-8: Correction of Field name “MAX_NUM” to “SequenceCount <= MAX_NUM”
		Section 10	Abbreviation XSM added

© 2010 OHB System AG

Table of Contents

© 2010 OHB System AG

© 2010 OHB System AG

© 2010 OHB System AG

© 2010 OHB System AG

List of Tables

© 2010 OHB System AG

List of Figures

© 2010 OHB System AG

1. INTRODUCTION

1.1 Purpose and Scope

This EPM Standard Protocol Definition has been established in order to describe the EPM internal standard protocols for the Command and Housekeeping data link and the LAN (Ethernet and USB).

The EPM specific file transfer is described in this document also, since the file transfer mechanism is standardized.

The specification for the Command and Housekeeping data link is applicable for the Facility internal communication between rack-mounted Science Modules (SM), the Science Module Support Computer (SMSC) and external Science Modules connected to EPM via the Utility Distribution Panel Interface that is also maintained by the SMSC.

The specification of the LAN protocol is applicable to internal communication between

- the Facility Control Computer (FCC) and the SMSC,
- the communication between the Laptop Unit (LTU) and the SMSC,
- the communication between the SM's and the SMSC,
- the Laptop internal communication between CLSW and SM-MMI
- the communication between the Carrier EGSE and Science Module EGSE.

The specification for USB communication is applicable to the onboard connection between SMSC and SM and the EGSE connection between Carrier EGSE and Science Module EGSE.

The specifications given in this document are applicable to all EPM models.

© 2010 OHB System AG

1.2 Overview

Figure 1-1 presents an overview of the Science Module to Facility Interfaces. Rack-mounted Science Modules can obtain their SAC location from the harness. The time synchronization is managed via the RS-485 interface. Time set via SNTP over LAN is provided for external Science Modules that are connected to the LAN on the Utility Distribution Panel.

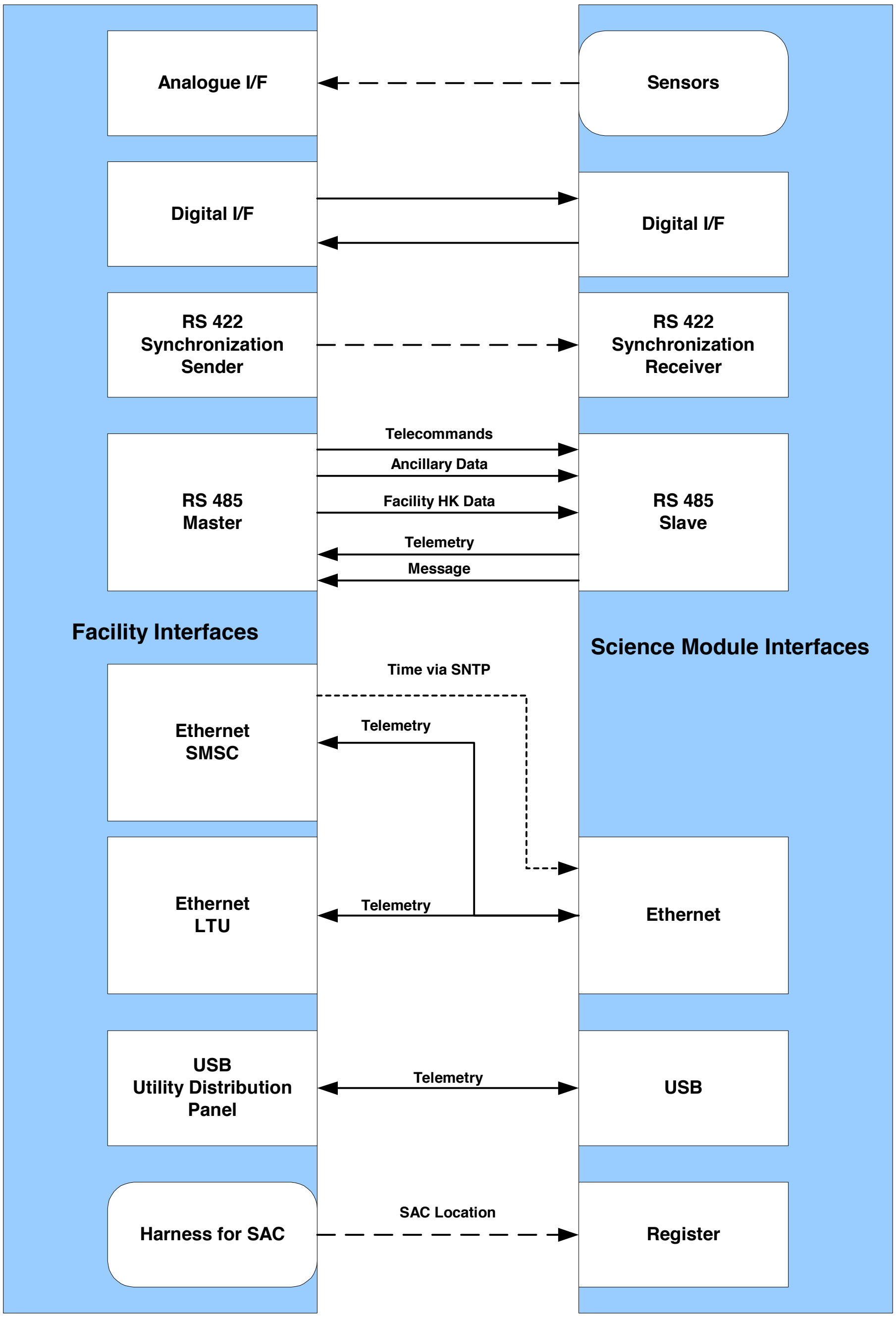

Figure 1-1: Overview of the on-board Science Module to Facility Interfaces

© 2010 OHB System AG

2. DOCUMENTS

2.1 Applicable Documents

This document shall be read in conjunction with the documents listed hereafter, which form part of this document to the extent specified herein. In case of a conflict between any provisions of this document and the provisions of the documents listed hereafter, the content of the contractually higher document shall be considered as superseding.

Doc. Ref.	*Doc. No.*	*Issue*	*Rev.*	*Title*
AD1	RFC 1769	March 1995	-	Simple Network Time Protocol (SNTP)
AD2	RFC 1305	March 1992	-	Network Time Protocol (NTP)
AD3	EPM-OHB-LI-0039	4	-	EPM Application Data TM/TC Data Definition Template
AD4	PI-MBUS-300	March 1993	E	Modicon Modbus Protocol Reference Guide
AD5		September 1998	1.1	Universal Serial Bus Specification Revision 1.1
AD6	CPI-RIBRE-ICD-0003-PIRN-0004	26.08.2004	1-	Columbus Ambient Sensor's Engineering Units

2.2 Reference Documents

Doc. Ref.	*Doc. No.*	*Issue*	*Rev.*	*Title*
RD1	EPM-OHB-MAN-0004	5	-	EPM LTU Environment and Services

© 2010 OHB System AG

3. STANDARDS

The following denominations are used for the bit field types:

- Nibble 4 bits
- Byte, Octet 8 bits
- Word 16 bits
- Double-Word 32 bits

For bit or byte rates

- k means: times 1024
- M means times 1048576 (1024 X 1024)

3.1 Agreements about Bit Numbering

In accordance with the ESA convention the following bit-numbering system is used:

- the MSB (most significant bit) has the bit number 0
- the LSB (least significant bit) has the bit number 7 (octet) or 15 (word) or 31 (double-word)

3.2 Coding of double words

Corresponding to the ESA Standards, double-words (32 bits) are transferred in the following way:

<table>
<tr><th>Bit # / Word #</th><th>MSB</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>LSB</th><th>Word Significance</th></tr>
<tr><td>1</td><td colspan="8">Octet 0 (MS Byte)</td><td colspan="8">Octet 1</td><td>High Word</td></tr>
<tr><td>2</td><td colspan="8">Octet 2</td><td colspan="8">Octet 3 (LS Byte)</td><td>Low Word</td></tr>
</table>

Byte 0 is the most significant byte within the 32-bit address, byte 3 the least significant byte.

The high word is transmitted first.

© 2010 OHB System AG

3.3 Coding of application data identifier

In order to provide a selection criterion and grouping of application data identifier all 16 bit identifier shall be coded in the following way:

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
ID	1	Identifier Group								ID in Identifier Group							

This allows for the MMI to implement a selection sequence:

Subsystem

Unit

Group

ID

For onboard, search algorithms can be avoided if group and ID's can be used directly as index of on-board tables.

This coding rule is applicable for:

- Telecommand identifier
- Telemetry identifier
- Message identifier
- Housekeeping and Status Data identifier

3.4 Format identification

Table 3-1 gives an overview of the format instructions for telecommand parameters and housekeeping data.

Format identification	Data type	Packet	Length in packet (octets)	Checks
li	long integer	TC/HK	4	optional: Range,

				Selection
i	integer	TC/HK	2	optional: Range, Selection
lu	long unsigned	TC/HK	4	optional: Range, Selection
u	unsigned	TC/HK	2	optional: Range, Selection
b	byte array	TC	even	optional: Size
f	float	TC/HK	4	optional: Range
d	double	TC/HK	8	optional: Range
s	alphanumeric	TC	even	optional: Size
t	time	TC	4	-

Table 3-1: Telecommand and housekeeping data parameter format instruction

For byte arrays and string the following rules are applicable:

The first two bytes give the actual length of the array/string. If the length is odd a fill byte (0x16) is appended. Strings must not be null-terminated.

Floating-point numbers use the IEEE (Institute of Electrical and Electronics Engineers) format.

The check instructions are:

- R for range
- L for selection from a list
- S for size

The values against which the check takes place are embedded in parentheses and separated by commas.

An example for the parameter description for the DOWNLINK FILE command is:

uL(0x0002, 0x0004, 0x0006) sS(1,212) sS(1,231)

where the first parameter indicates the telemetry destination (FRC, USOC, FRC & USOC) and the two strings indicate the filenames of the sender and the receiver.

Note: The parameter instructions are the same for IAP instructions and the manual commanding menu of the CLSW. However, commas separate individual parameters, strings do not contain the length and are embedded in quotation marks. E.g. the user input to CLSW or inside an IAP for a file ‘c:\test.dat’ to be downlinked to FRC into ‘x:\dumpdata.dat’ is: u2, s”c:\test.dat”, s”x:\dumpdata.dat”.

© 2010 OHB System AG

3.5 Model identification

The identification of models is introduced in order to allow relating data to the environmental conditions under which they have been recorded.

Model	Model ID
FM	00000001/b
GM-1	00000010/b
GM-2	00000100/b
BDCM	00001000/b
EAC-TRM	00010000/b

Table 3-2: Model ID’s

© 2010 OHB System AG

3.6 Subsystem identification

The application data for each subsystem will be identified by an 8 bit subsystem ID. Table 3-3 provides the list of subsystem ID's for EPM. The Set subsystem Time command is sent to all Science Modules via RS-485 interface via broadcast and thus uses the Subsystem ID "ALL".

Subsystem	Sub system ID
ALL (All Science Modules)	0
SAC 1 (D1)	11
SAC 2 (E1)	12
SAC 3 (F1)	13
SAC 4 (G1)	14
SAC 5 (D2)	21
SAC 6 (E2)	22
SAC 7 (F2)	23
SAC 8 (G2)	24
External Science Modules connected to the Utility Distribution Panel	31 – 41
FCC	1
SMSC	2
LTU (Utility Distribution Panel)	3
VU	30

Table 3-3: Subsystem ID's

If a Science Module occupies more than one position only the ID of the position which is used to electrically connect to EPM shall be used.

3.7 Operational mode

The mode shall be coded in one octet where each bit identifies a specific mode according to the following definition:

Mode	EPM Mode Coding	Columbus Converted Value
Device inactive	10000000/b	*0*
Startup	00000001/b	*1*
Check-out	00000010/b	*2*
Wait_on_sync	00000100/b	*3*
Setup	00001000/b	*4*
Nominal	00010000/b	*5*
Test	00100000/b	*6*
Unknown	01000000/b	*7*

Table 3-4: S/W Modes

© 2010 OHB System AG

The “Device inactive” mode signals that the device is not activated and thus is not ready for operation. The value is set by the unit that acquires the mode of the device.

This coding allows coding of mode validity for housekeeping check instruction and telecommand execution admission in one byte. E.g. all permanent modes (these are all modes except the start up mode) will be coded as a mask: 00111110/b.

3.8 Timer Status

The timer status shall be coded in one octet where each bit identifies a specific status value according to the following definition:

Bit #	Value	Bit Set	Bit Not Set
7 (LSB)	Time Set Status	Time Set	Time Not Set
6	Time set via time set command	Time Set	Time Not Set
5	Time set via NTP	Time Set	Time Not Set
4	Time based on sync pulse	Yes	Pulse not available
3	Time based on internal timer	Yes	No
2	SM Elapsed time after activation	Yes	No
1	Not used	-	-
0 (MSB)	Not used	-	-

Table 3-5: Timer Status

Unless the time is not set at all, the SM time is only the elapsed time after activation, starting from 0 seconds based on internal timer.

After activation the Science Module timer status is thus 00110000/b. EPM evaluates the timer status and initiates the time set procedure according to the current configuration. For a rack-mounted Science Module in nominal configuration the time will be set via command and the synchronization pulse will be the basis for the SM time maintenance, thus the timer status will become 0001011/b.

3.9 Checksum control

Different checksums are possible for EPM in order to ensure, that controllers that have low performance can use Vertical Parity Check instead of CRC or even a simple EOT marker instead.

Checksum	Coding
Fixed bit pattern (EOT)	00/b
Vertical Parity Check	01/b
CRC	10/b

Table 3-6: Checksum Control

© 2010 OHB System AG

For EOT the fixed pattern is:

Byte	0	1
Hex	03	04
Bin	00000011	00000100

VPC (Vertical Paritiy Check) and CRC (Cyclical Redundancy Check) are 16 bit checksum of unsigned integer type. The checksum algorithm is a 16-bit checksum over the complete structure exclusive the checksum field itself.

The VPC calculation is done a s follows:

1. Set Checksum to zero
2. Start with first byte of data block
3. New Checksum is Old Checksum Exclusive OR Current Byte of Data Block
4. Increment byte position in Data block
5. Go to step 3 until last byte of data block is reached

The step by step procedure to form the CRC-16 is as follows:

1.Load a 16-bit register with all 1's.

2.Exclusive OR the first 8-bit byte with the high order byte of the 16-bit register, putting the result in the 16-bit register.

3.Shift the 16-bit register one bit to the right.

4a.If the bit shifted out to the right is one, exclusive OR the generating polynomial 1010 0000 0000 0001 with the 16-bit register.

4b.If the bit shifted out to the right is zero; return to step 3.

5.Repeat steps 3 and 4 until 8 shifts have been performed.

6.Exclusive OR the next 8-bit byte with the 16-bit register.

7.Repeat step 3 through 6 until all bytes of the message have been exclusive OR'rd with the 16-bit register and shifted 8 times.

8.The contents of the 16-bit register are the 2 byte CRC error check and is added to the message most significant bits first.

© 2010 OHB System AG

An example of the CRC-16 error check for message HEX 0207:

	16-BIT REGISTER			MSB	Flag
(Exclusive OR)	1111	1111	1111	1111	
02			0000	0010	
	1111	1111	1111	1101	
Shift 1	0111	1111	1111	1110	1
Polynomial	1010	0000	0000	0001	
	1101	1111	1111	1111	
Shift 2	0110	1111	1111	1111	1
Polynomial	1010	0000	0000	0001	
	1100	1111	1111	1110	
Shift 3	0110	0111	1111	1111	0
Shift 4	0011	0011	1111	1111	1
Polynomial	1010	0000	0000	0001	
1001	0011	1111	1110		
Shift 5	0100	1001	1111	1111	0
Shift 6	0010	0100	1111	1111	1
Polynomial	1010	0000	0000	0001	
	1000	0100	1111	1110	
Shift 7	0100	0010	0111	1111	0
Shift 8	0010	0001	0011	1111	1
Polynomial	1010	0000	0000	0001	
	1000	0001	0011	1110	
07			0000	0111	
	1000	0001	0011	1001	
Shift 1	0100	0000	1001	1100	1
Polynomial	1010	0000	0000	0001	
	1110	0000	1001	1101	
Shift 2	0111	0000	0100	1110	1
Polynomial	1010	0000	0000	0001	
	1101	0000	0010	1111	
Shift 3	0110	1000	0010	0111	1
Polynomial	1010	0000	0000	0001	
	1100	1000	0010	0110	
Shift 4	0110	0100	0001	0011	0
Shift 5	0011	0010	0000	1001	1
Polynomial	1010	0000	0000	0001	
	1001	0010	0000	1000	
Shift 6	0100	1001	0000	0100	0
Shift 7	0010	0100	1000	0010	0
Shift 8	0001	0010	0100	0001	0
	HEX 12		HEX 41		

TRANSMITTED MESSAGE WITH CRC-16
(MESSAGE SHIFTED TO RIGHT TO TRANSMIT)

12		41		07		02	
0001	0010	0100	0001	0000	0111	0000	0010

© 2010 OHB System AG

3.10 Command source

The coding of the command source follows the same philosophy as the coding for modes.

Command Source	Command Source Coding
Columbus CC*	00000001/b
FRC	00000010/b
USOC	00000100/b
FCC	00010000/b
SMSC	00100000/b
LTU	01000000/b
HRF*	10000000/b

Table 3-7: Command Sources

*Only the Carrier DMS can execute Commands from Columbus CC and HRF (to be defined on a case by case basis).

3.11 Command priority

The command priority controls the execution priority.

Command Priority	Command Priority Coding
Standard	00/b
High priority	01/b
Urgent priority	10/b

Table 3-8: Command Priorities

3.12 Sequence flags

The following definition is applicable to any sequenced data transfer.

Sequence Flag	Sequence Flag Coding
Continue	00/b
First packet	01/b
Last packet	10/b
Standalone packet	11/b

Table 3-9: Sequence Flags

© 2010 OHB System AG

3.13 Telemetry destination

The telemetry destination specifies the receiver of the telemetry package.

Telemetry Destination	Telemetry Destination Coding
FRC	00000010/b
USOC	00000100/b
Internal	00001000/b
FCC	00010000/b
SMSC	00100000/b
LTU	01000000/b
HRF	10000000/b

Table 3-10: Telemetry Destination

The telemetry destination is set to internal if two nodes inside the facility exchange data directly (e.g. internal file transfer between to SMSC and SM).

The receiver SubSystem ID and Unit ID is used to determine the receiver if telemetry destination is set to internal.

3.14 DELETED

3.15 S/W Version

The software version shall be coded in one word.

It is used to ensure that the SM-MMI and the SM software are always consistent.

Each Version is identified by the following system:

aa.bb.cc/xx

xx identifies the verification state,

- 0: for all incremental releases during development and until all modules are available
- alpha: for all incremental releases during development and until at least all tests on CSCI level have been performed
- beta: for all releases until all facility level tests have been performed
- " ": for acceptance of EPM-1

For major changes on the original SW during OM phase, e.g. adding new interface modules due to changes in the EPM hardware the number shall be incremented.

aa, bb and cc identify the version for the model and CSCI.

© 2010 OHB System AG

- aa is incremented for major changes, e.g. adding a module, changing module external interfaces
- bb is incremented for medium changes, like module internal functional changes
- cc is incremented for minor changes, like module internal minor changes

A SW release includes all command, message, display and HK-data definitions, the documentation and test procedures.

Version Field	Nibble
aa	3
bb	2
cc	1
xx	0

Table 3-11: Version Coding

Verification State	Value
0	0001/b
Alpha	0010/b
Beta	0100/b
Verified and accepted	1000/b

Table 3-12: Verification State coding

Note: The verification state coding allows checking for valid entries by the condition that the number of set bit is one.

© 2010 OHB System AG

3.16 Housekeeping Value Check Status

The housekeeping value check status is an octet that codes the runtime check results. It is used to regulate the displays on the MMI, e.g. color change in case of non-nominal values.

Note: HK checks are applied to HK-Value/Mask combination. E.g., checks of individual bits of status registers are supported. In this case the entire register is placed into the Bulk Housekeeping Data Report Telemetry Packet but several checks can be applied to the register.

Different checks can be applied for different operational modes. The mode information in the header of the telemetry packet indicates the operational mode at the time the packet has been generated. The housekeeping value check status is not affected, if a mode change has occurred between the HK-data set provision and the packet generation.

Bit #	Value	Bit Set	Bit Not Set
7 (LSB)	HK Check Status	At least one check has failed	All checks successful
6	Warning	Report as warning	No warning report
5	Error	Report as error	No error report
4	Message Generation	Enabled	Disabled
3	Compare Check	Compare Check has failed	Ok
2	Minimum	Minimum check failed	Ok
1	Maximum	Maximum check failed	Ok
0 (MSB)	Data Validity	Data Valid	Data Invalid

Table 3-13: Housekeeping Value Check Status

The housekeeping check status list is part of each housekeeping data report.

The Data Validity can be used to signal that data are timeout (no new data available) or not acquired in the current operational mode.

For a minimum violation of a valid analogue value that has been reported as warning, where message generation is still active, the housekeeping value check status is 10101011/b.

3.17 Data alignment and fill bytes

According to AD 1, any EPM telecommand shall be aligned to words, any telemetry shall be aligned to double words and any internal structure shall be aligned to double words.

If fill bytes are required, these bytes shall be set to 0x16.

© 2010 OHB System AG

4. LAN INTERFACE

The rules for the LAN are applicable for Ethernet between all nodes and USB interface between VU and LTU and SM and SMSC.

4.1 IP Adresses and Port number

The addresses are related to the corresponding SAC position and coded in the harness.

EPM uses Class C Network Numbers. IP addresses are composed of a Network part (indicated by NNN) and a Host part (indicated by HHH). EPM addresses are thus represented by:

NNN.NNN.NNN.HHH

The Network part of EPM Ethernet is set to:

192.168.200

The Network part of EPM USB is set to:

192.168.201

Table 4-1 lists the Host-part of internal IP addresses for the EPM LAN.

Location	Host part of IP Address
SAC 1 (D1)	111
SAC 2 (E1)	112 (67, see below)
SAC 3 (F1)	113
SAC 4 (G1)	114
SAC 5 (D2)	121
SAC 6 (E2)	122 (77, see below)
SAC 7 (F2)	123
SAC 8 (G2)	124
Utility Distribution Panel 1	131
Utility Distribution Panel 2	132
FCC	101
SMSC	102
Utility Distribution Panel (LTU)	103
VU	130
EGSE	200-254

Table 4-1: Host part of LAN IP addresses

As an example, the Ethernet IP Address of a SM accommodated in SAC position D1 thus becomes:

192.168.200.111

© 2010 OHB System AG

If a Science Module occupies more than one position only the IP of the position which is used to electrically connect to EPM shall be used.

The Port number for all EPM LAN connections is **2345**.

Remark: During startup pre-configured default IP addresses are used by Rackmounted Science Modules for a short time, e.g. 67 (Cardiolab) and 77 (MEEMM). After startup these addresses are changed in accordance to Table 4-1.

4.2 TCP/IP SW-Unit Addressing

The TCP/IP protocol will be used for the communication over the EPM LAN and for data exchange between CLSW and SM-MMI.

If data transferred to the LTU shall be transferred to ground also, then these data have to be sent twice to the different addressees by the SM.

Figure 4-1 provides an overview of the TCP/IP server and clients in EPM.

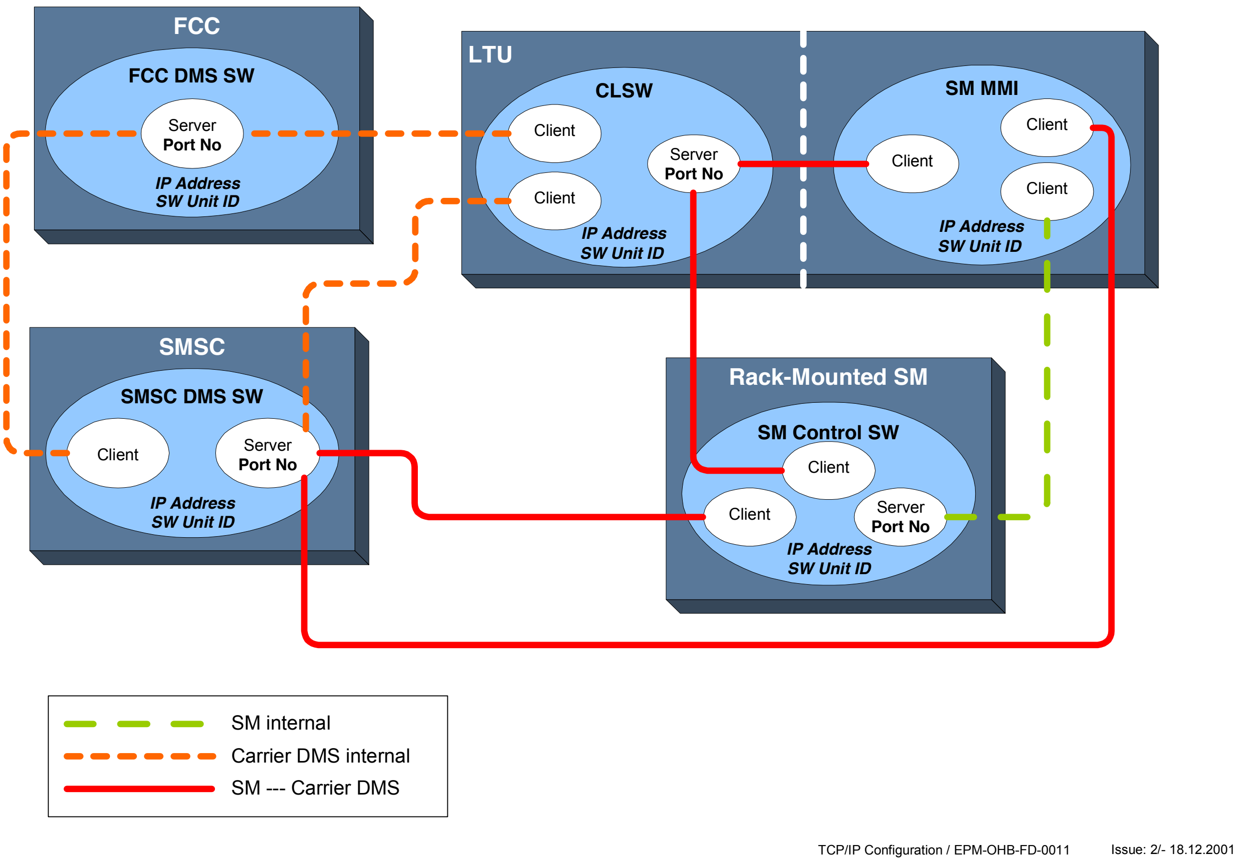

Figure 4-1: Internal TCP/IP Client and Server Overview

The SMSC runs a TCP/IP server to which a client can connect. SM's shall run a client that allows connection to this server. SM-MMI's can also connect to this server.

The SM shall furthermore run a server. An SMM-MMI client connects to this server to allow direct data exchange between SM and SM-MMI.

The CLSW runs a server to which a SM-MMI client can connect. The CLSW client is used for connection to the Carrier DMS server.

The SM-MMI receives all TM packets of the system that are transferred to the LTU (destination field flag LTU is set) except file transfer TM and MPEG and JPEG TM that are processed by the CLSW only.

The number of simultaneously open client connections to the Carrier DMS and the LTU is limited to 10, respectively.

The Software unit identification specifies the EPM unique software unit.

If a Science Module occupies more than one position only the IDs of the position which is used to electrically connect to EPM shall be used.

Note: The subsystem ID and the subsystem unit ID are device identifier. That means they identify hardware components. The Software Unit Identifier uniquely identifies software programs or modules. The SW Unit ID is used on TCP/IP level only in order to control TCP/IP connections and to control the application data distribution

Related Device	Software Unit ID Ethernet	Software Unit ID USB	Software Unit ID MMI
SAC 1 (D1)	11	111	41
SAC 2 (E1)	12	112	42
SAC 3 (F1)	13	113	43
SAC 4 (G1)	14	114	44
SAC 5 (D2)	21	121	51
SAC 6 (E2)	22	122	52
SAC 7 (F2)	23	123	53
SAC 8 (G2)	24	124	54
External SM (1..10)	31-40	131-140	61-70
FCC	1	-	-
SMSC	2	102	-
CLSW	3	103	-
VU	-	130	Ethernet:60, USB 160
EGSE	200-218	219-236	237-254

Table 4-2: Software unit identification

© 2010 OHB System AG

4.3 TCP/IP Transfer Frame

The TCP/IP transfer frame is shown in Table 4-3.

The Software Unit ID is actually a byte.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
EPM LAN Prot Head.	1	EPM LAN Sync Marker (Word #1 0xAA49)															
	2	EPM LAN Sync Marker (Word #2 0xDBFF)															
	3	Spare = 0								Software Unit ID							
	4	Packet Type															
	5	Spare															
	6	Number of Transferred Words															
App. Data	7																
	M																

Table 4-3: Application Data transfer frame for TCP/IP

4.3.1 EPM LAN Sync Marker

The EPM-LAN synchronization marker is a double word. It is set to:

Byte	0	1	2	3
Hex	AA	49	DB	FF
Bin	10101010	01001001	11011011	11111111

4.3.2 Software Unit ID

The Software Unit ID specifies the EPM unique software unit according to Table 4-2.

The ID of the sender is inserted in the header.

4.3.3 Packet Type

This field specifies the type of the transferred packet. These are:

- Protocol specific types

© 2010 OHB System AG

- Application specific types
- SM-MMI control specific types

Protocol specific types are:

- **Connect** **0x0001**
  Science Modules can send this type of packet in order to supply the server with additional information about the connecting entity.

- **Alive** **0x0002**
  Clients maintaining a permanent connection to a server have to send this type of packet in regular intervals to inform the server that the client side is still in a fully functional state. Alive packets have only to be sent when no other traffic occurs on the peer-to-peer connection.
  The interval between ALIVE Packets must not exceed two seconds but at least be one second.

Application specific types are:

- **Telecommand** **0x1154**
- **Telemetry** **0x1153**
- **HRD Bitsteam** **0x2053** (only used from HRDReceiver and HDRConverter)

SM-MMI control specific types are:

- **Directive** **0xBB44**
- **Directive Acknowledge** **0xBB06**
- **Setting Instruction** **0xBB49**
- **Procedure Message** **0xBB50**

4.3.4 Spare

The spare is not used. It is set to zero

4.3.5 Number of transferred Words

This field gives the number of total words that are transferred.

The amount of user data within one packet shall be such that the resulting TCP/IP packet fits into a single Ethernet frame, i.e. no segmentation in several Ethernet frames shall be

necessary. This means that the maximum amount of user data is limited to 1412 octets

including the necessary amount of octets for the application headers (EPM LAN Protocol header, TM/TC headers).

TM/TC packets must be embedded within one Ethernet frame, and it is only allowed to transmit one TM/TC packet within an Ethernet frame.

© 2010 OHB System AG

4.3.6 Application data

The application data are a TC, TM or control data for application specific types.

TC and TM follow the layout as described in section 7.

Control data are used for data exchange between CLSW and SM-MMI. The SM-MMI control specific types structures are described in section 8.

4.4 Time Set via SNTP

EPM supports time set via LAN using SNTP. The SMSC runs the SNTP server and the LTU (CLSW) requests the time. The time of a SM can also be set via that service. However, the use of RS-422 time sync together with the RS-485 command is mandatory for rack-mounted Science Modules. External Science Modules connected via LAN to the Utility Distribution Panel only can use the service.

The SMSC provides an SNTP server for distribution of time information via LAN. Science

Modules can connect to the server and request an SNTP timestamp packet, according to the specifications defined in RFC 1769.

Related standards are documented in:

Simple Network Time Protocol (SNTP): RFC 1769, March 1995

Network Time Protocol (NTP): RFC 1305, March 1992

© 2010 OHB System AG

5. RS 485 PROTOCOL

The command and housekeeping interface for rack-mounted Science Modules is the RS485 link. All telecommands and the EPM ancillary data are transferred via that link to the SM's and any system information data is transferred to the Carrier DMS over this interface.

The ModBusProtocol (AD 4) using RTU (Remote Terminal Unit) Mode is used.

The settings are:

- Baud rate: 9600 Baud
- Parity: No parity
- 2 Stopbits

5.1 RS 485 Addresses

The RS-485 addresses for the different positions are listed in Table 5-1.

Location	Address
Broadcast	**0**
SAC 1 (D1)	11
SAC 2 (E1)	12
SAC 3 (F1)	13
SAC 4 (G1)	14
SAC 5 (D2)	21
SAC 6 (E2)	22
SAC 7 (F2)	23
SAC 8 (G2)	24
VU	30
Utility Distribution Panel 1	31
Utility Distribution Panel 2	32

Table 5-1: RS 485 addresses

If a Science Module occupies more than one position only the address of the position used to electrically connect to EPM shall be used.

5.2 MODBUS Protocol

This protocol defines a message structure that controllers will recognize and use, regardless of the type of networks over which they communicate. It describes the process a controller uses to request access to another device, how it will respond to requests from the other devices, and how errors will be detected and reported. It establishes a common format for the layout and contents of message fields.

Controllers communicate using a master-slave technique, in which only one device (the master) can initiate transactions (queries). The other devices (the slaves) respond by supplying the requested data to the master, or by taking the action requested in the query.

© 2010 OHB System AG

Typical master devices include host processors and programming panels. Typical slaves include programmable controllers.

The master can address individual slaves, or can initiate a broadcast message to all slaves. Slaves return a message (response) to queries that are addressed to them individually. Responses are not returned to broadcast queries from the master.

The Modbus protocol establishes the format for the master's query by placing into it the device (or broadcast) address, a function code defining the requested action, any data to be sent, and an error-checking field. The slave's response message is also constructed using Modbus protocol. It contains fields confirming the action taken, any data to be returned, and an error-checking field. If an error occurred in receipt of the message, or if the slave is unable to perform the requested action, the slave will construct an error message and send it as its response.

The function code in the query tells the addressed slave device what kind of action to perform. The data bytes contain any additional information that the slave will need to perform the function. For example, function code 03 will query the slave to read holding registers and respond with their contents. The data field must contain the information telling the slave which register to start at and how many registers to read. The error check field provides a method for the slave to validate the integrity of the message contents.

If the slave makes a normal response, the function code in the response is an echo of the function code in the query. The data bytes contain the data collected by the slave, such as register values or status. If an error occurs, the function code is modified to indicate that the response is an error response, and the data bytes contain a code that describes the error. The error check field allows the master to confirm that the message contents are valid.

5.2.1 Data encoding

MODBUS uses a 'big-endian' representation for addresses and data items. This means that when a numerical quantity larger than a single byte is transmitted, the MOST significant byte is sent first. So for example

16 - bits	0x1234	would be	0x12	0x34		
32 - bits	0x12345678L	would be	0x12	0x34	0x56	0x78

5.2.2 RTU Framing

In RTU mode, messages start with a silent interval of at least 3.5 character times. This is most easily implemented as a multiple of character times at the baud rate that is being used on the network (shown as T1-T2-T3-T4 in the figure below). The first field then transmitted is the device address.

The allowable characters transmitted for all fields are hexadecimal 0 ... 9, A ... F. Networked devices monitor the network bus continuously, including during the silent intervals. When the first field (the address field) is received, each device decodes it to find out if it is the addressed device.

Following the last transmitted character, a similar interval of at least 3.5 character times marks the end of the message. A new message can begin after this interval.

The entire message frame must be transmitted as a continuous stream. If a silent interval of more than 1.5 character times occurs before completion of the frame, the receiving device flushes the incomplete message and assumes that the next byte will be the address field of a new message.

Similarly, if a new message begins earlier than 3.5 character times following a previous message, the receiving device will consider it a continuation of the previous message. This will set an error, as the value in the final CRC field will not be valid for the combined messages. A typical message frame is shown below.

T1,T2,T3	ADDRESS	FUNCTION	DATA	CHECK	T1,T2,T3
	8-BITS	8-BITS	N X 8-BITS	16-BITS	

Frame synchronization can be maintained in RTU transmission mode only by simulating a synchronous message. The receiving device monitors the elapsed time between receipt of characters. If three and one-half character times elapse without a new character or completion of the frame, then the device flushes the frame and assumes that the next byte received will be an address.

5.2.3 Address Field

The address field of a message frame contains eight bits. Valid slave device addresses are in the range of 0 ... 247 decimal. The individual slave devices are assigned addresses in the range of 1 ... 247. A master addresses a slave by placing the slave address in the address field of the message. When the slave sends its response, it places its own address in this address field of the response to let the master know which slave is responding.

The address field immediately follows the beginning of frame and consists of 8-bits. These bits indicate the user assigned address of the slave device that is to receive the message sent by the attached master.

Each slave must be assigned a unique address and only the addressed slave will respond to a query that contains its address. When the slave sends a response, the slave address informs the master which slave is communicating. In a broadcast message, an address of 0 is used. All slaves interpret this as an instruction to read and take action on the message, but not to issue a response message.

5.2.4 Function Field

The Function Code field tells the addressed slave what function to perform

The function code field of a message frame contains eight bits. Valid codes are in the range of 1 ... 255 decimal. Of these, some codes are applicable to all controllers, while some codes apply only to certain models, and others are reserved for future use.

When a message is sent from a master to a slave device the function code field tells the slave what kind of action to perform. Examples are to read the ON / OFF states of a group of discrete coils or inputs; to read the data contents of a group of registers; to read the diagnostic status of the slave; to write to designated coils or registers; or to allow loading, recording, or verifying the program within the slave.

When the slave responds to the master, it uses the function code field to indicate either a normal (error-free) response or that some kind of error occurred (called an exception response). For a normal response, the slave simply echoes the original function code. For an exception response, the slave returns a code that is equivalent to the original function code with its most significant bit set to a logic 1.

For example, a message from master to slave to read a group of holding registers would have the following function code:

0000 0011 (Hexadecimal 03)

If the slave device takes the requested action without error, it returns the same code in its response. If an exception occurs, it returns:

1000 0011 (Hexadecimal 83)

In addition to its modification of the function code for an exception response, the slave places a unique code into the data field of the response message. This tells the master what kind of error occurred, or the reason for the exception.

The master device's application program has the responsibility of handling exception responses. Typical processes are to post subsequent retries of the message, to try diagnostic messages to the slave, and to notify operators.

5.2.5 Data Field

The data field contains information needed by the slave to perform the specific function or it contains data collected by the slave in response to a query.

The data field is constructed using sets of two hexadecimal digits, in the range of 00 to FF hexadecimal.

The data field of messages sent from a master to slave devices contains additional information which the slave must use to take the action defined by the function code. This can include items like discrete and register addresses, the quantity of items to be handled, and the count of actual data bytes in the field.

For example, if the master requests a slave to read a group of holding registers (function code 03), the data field specifies the starting register and how many registers are to be read. If the master writes to a group of registers in the slave (function code 10 hexadecimal), the data field specifies the starting register, how many registers to write, the count of data bytes to follow in the data field, and the data to be written into the registers.

If no error occurs, the data field of a response from a slave to a master contains the data requested. If an error occurs, the field contains an exception code that the master application can use to determine the next action to be taken.

The data field can be nonexistent (of zero length) in certain kinds of messages. For example, in a request from a master device for a slave to respond with its communications

© 2010 OHB System AG

event log (function code 0B hexadecimal), the slave does not require any additional information. The function code alone specifies the action.

5.2.6 Parity Checking

With RTU character framing, the bit sequence is:

With Parity Checking

Start	1	2	3	4	5	6	7	8	Par	Stop

Without Parity Checking

Start	1	2	3	4	5	6	7	8	Stop	Stop

Users can configure controllers for Even or Odd Parity checking, or for No Parity checking. This will determine how the parity bit will be set in each character.

If either Even or Odd Parity is specified, the quantity of 1 bits will be counted in the data portion of each character (eight for RTU). The parity bit will then be set to a 0 or 1 to result in an Even or Odd total of 1 bits. For example, these eight data bits are contained in an RTU character frame:

1100 0101

The total quantity of 1 bits in the frame is four. If Even Parity is used, the frame's parity bit will be a 0, making the total quantity of 1 bits still an even number (four). If Odd Parity is used, the parity bit will be a 1, making an odd quantity (five).

When the message is transmitted, the parity bit is calculated and applied to the frame of each character. The receiving device counts the quantity of 1 bits and sets an error if they are not the same as configured for that device (all devices on the Modbus network must be configured to use the same parity check method).

Note that parity checking can only detect an error if an odd number of bits are picked up or dropped in a character frame during transmission. For example, if Odd Parity checking is employed, and two 1 bits are dropped from a character containing three 1 bits, the result is still an odd count of 1 bits.

If No Parity checking is specified, no parity bit is transmitted and no parity check can be made. An additional stop bit is transmitted to fill out the character frame.

EPM implementation is without parity checking.

5.2.7 CRC-16 Cyclic Redundancy Check

The CRC-16 error check sequence is implemented as described in section 3.9.

© 2010 OHB System AG

5.2.8 Supported Data and Control functions

Not supported by SMSC

5.2.9 Supported Diagnostic Subfunctions

Not supported by SMSC

5.2.10 Supported Exception Responses

Not supported by SMSC

5.3 Extension to MODBUS specification for Function Code 0x41

The function code 0x41 is used for application data transfer.

The MODBUS implied length philosophy defines that all MODBUS requests and responses are designed in such a way that the recipient can verify that a message is complete. For function codes where the request and response are of fixed length, the function code alone is sufficient. For function codes carrying a variable amount of data in the request or response, the data portion will be preceded by a byte count.

The standard MODBUS implementation thus limits the length of messages to 256. Since application data have length up to 1400 bytes the standard length philosophy is extended to packages up to this size for function code 0x41.

It is possible that the SM has to transfer telecommands to external SM devices and thus acts as TC distributor on SM level. Since the timeout for start of the response by the slave device is set to 0.1 seconds it is not required that the response contain data generated by the last telecommand.

After receipt of a telecommand that is always transferred together with the mandatory ancillary data, the SM responds with the mandatory critical housekeeping data set and application data if available.

It is not required that the user data transferred belong to the telecommand.

Table 5-2 shows the RTU Frame for Function Code 0x41.

<table>
<tr><th>Field ID</th><th>Bit # / Word #</th><th>MSB</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>LSB</th></tr>
<tr><td rowspan="2">Prot Head.</td><td>1</td><td colspan="8">Slave Address</td><td colspan="8">Function Code = 0x41</td></tr>
<tr><td>2</td><td colspan="16">Length in Bytes</td></tr>
<tr><td rowspan="4">User Data</td><td>3</td><td colspan="16"></td></tr>
<tr><td></td><td colspan="16"></td></tr>
<tr><td></td><td colspan="16"></td></tr>
<tr><td>N-2</td><td colspan="16"></td></tr>
<tr><td>Check</td><td>N-1</td><td colspan="16">CRC</td></tr>
</table>

Table 5-2: Modbus RTU Frame for Function Code 0x41

© 2010 OHB System AG

The length in bytes counts the user data bytes.

It is the message length –6.

© 2010 OHB System AG

5.4 Application Data transfer frame Carrier DMS to SM

All Spare Fields are set to 0.

5.4.1 Transfer Frame for Broadcast Messages

Table 5-3 describes the application data transfer frame for data transfer from the Carrier DMS for broadcast messages.

Note SMs do not response upon receipt of a broadcast message. The Set Time Telecommand and the Ancillary Data are provided via Broadcast.

<table>
<tr><th>Field ID</th><th>Bit # / Word #</th><th>MSB</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>LSB</th></tr>
<tr><td rowspan="2">CTRL Head.</td><td>1</td><td colspan="8">Spare 0</td><td colspan="8">Type of Data</td></tr>
<tr><td>2</td><td colspan="16">Spare 1</td></tr>
<tr><td rowspan="4">Mand. Anc. Data</td><td>3</td><td colspan="8">Contact Status</td><td colspan="8">Laptop Status</td></tr>
<tr><td>4</td><td colspan="8">Spare 2</td><td colspan="8">Sync. Pulse Status</td></tr>
<tr><td>5</td><td colspan="16">Spare 3</td></tr>
<tr><td>6</td><td colspan="16">Spare 4</td></tr>
<tr><td rowspan="3">App. Data</td><td>7</td><td colspan="16"></td></tr>
<tr><td></td><td colspan="16"></td></tr>
<tr><td>M</td><td colspan="16"></td></tr>
</table>

Table 5-3: Application Data transfer frame for broadcast messages from Carrier DMS

5.4.2 Transfer Frame for Individual Messages

Table 5-4 provides an overview of the application data transfer frame for data transfer from the Carrier DMS to an individual SM.

© 2010 OHB System AG

<table>
<tr><th>Field ID</th><th>Bit # / Word #</th><th>MSB</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>LSB</th></tr>
<tr><td rowspan="2">CTRL Head.</td><td>1</td><td colspan="8">Previous app. data packet ack.</td><td colspan="8">Type of Data</td></tr>
<tr><td>2</td><td colspan="16">Spare 0</td></tr>
<tr><td rowspan="4">Mand. Anc. Data</td><td>3</td><td colspan="8">Contact Status</td><td colspan="8">Laptop Status</td></tr>
<tr><td>4</td><td colspan="8">Spare 1</td><td colspan="8">Sync. Pulse Status</td></tr>
<tr><td>5</td><td colspan="16">RS 485 unique TC packet counter</td></tr>
<tr><td>6</td><td colspan="16">Spare 2</td></tr>
<tr><td rowspan="3">App. Data</td><td>7</td><td colspan="16"></td></tr>
<tr><td></td><td colspan="16"></td></tr>
<tr><td>M</td><td colspan="16"></td></tr>
</table>

Table 5-4: Application Data transfer frame for individual slaves from Carrier DMS to SM

5.4.3 Previous Application Data packet acknowledge

The field is only used if an individual Science Module is addressed. It is not used for broadcast messages.

This field provides information of the receipt status of the previous package received from the SM. The SM shall evaluate this field.

Values are:

- 0x15 for NACK, data not received properly
- 0x06 for ACK, data received properly
- 0x18 for Transfer acknowledge, but Application Data not processed

If the value is not set to ACK, the SM shall transmit the application data of the previous request.

5.4.4 Type of Data

This field provides information of the application data type of this package: Values are:

- 0x00, for No Data

© 2010 OHB System AG

- 0x43, for telecommand
- 0x41, for ancillary data

5.4.5 Contact Status

This field provides information on the ground contact status: Values are:

- 0x00, no contact (LOS)
- 0x0F, contact (AOS S-Band)
- 0xF0, contact (AOS Ku-Band)
- 0xFF, contact (AOS S-Band and Ku-Band)

5.4.6 Laptop Status

This field provides information on the Laptop status: The Carrier DMS controls whether the LTU is running by a watchdog request algorithm similar to the implementation on the RS 485 protocol and provides this value to the SM's. The SM can react on this status information, e.g. suspension of a SM protocol. Values are:

- 0x00, Laptop not connected
- 0xFF, Laptop connected

5.4.7 Sync Pulse Status

This field provides information on the current use of the RS-422 sync pulse. Values are:

- 0x00, pulse used for time synchronization
- 0xFF, pulse used for experiment synchronization

5.4.8 RS 485 unique TC counter

The field is only used if an individual Science Module is addressed. It is not used for broadcast messages.

This field contains a unique counter for each telecommand that is sent. If the Carrier DMS cannot receive/evaluate the response package of the SM, the command is sent again with the same counter. This allows that the SM can detect whether the command has been received already.

The Carrier DMS deletes the command and generates a command reject message after three failed attempt to transfer the command.

Note: The counter is incremented by the Carrier DMS for any telecommand. It remains unchanged if application data are NO-DATA.

5.4.9 Application Data

These fields are not present if the data type is No Data or a telecommand if the data Type is set to TC or ancillary data if Type is set to Ancillary Data.

5.4.9.1 Data Structure for application Data

5.4.9.1.1 Ancillary Data (Type 0x41)

Ancillary data are taken over from Columbus and provided to the SM.

The structure is defined in section 7.

5.4.9.1.2 Telecommand (Type 0x43)

The telecommand structure is defined in section 7.

© 2010 OHB System AG

5.5 Application Data transfer frame SM to Carrier DMS

Table 5-5 provides an overview of the application data transfer frame for data transfer from the SM to the Carrier DMS.

Note this frame is not sent if a broadcast message has been received.

All Spare Fields are set to 0.

<table>
<tr><th>Field ID</th><th>Bit # / Word #</th><th>MSB</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>LSB</th></tr>
<tr><td rowspan="2">CRTL Head.</td><td>1</td><td colspan="8">Previous app. data packet acknowledge</td><td colspan="8">Type of Data</td></tr>
<tr><td>2</td><td colspan="16">Spare 1 = 0</td></tr>
<tr><td rowspan="8">Mand. Crit. Hk. Data</td><td>3</td><td colspan="8">SM Timer Status</td><td colspan="8">SM Mode</td></tr>
<tr><td>4</td><td colspan="8">Number of SM Messages</td><td colspan="8">Number of SM TM Packets</td></tr>
<tr><td>5</td><td colspan="8">Spare 2 = 0</td><td colspan="8">SM HK Acquisition Rate</td></tr>
<tr><td>6</td><td colspan="16">Current SM Process ID</td></tr>
<tr><td>7</td><td colspan="16">Application data Rate LAN</td></tr>
<tr><td>8</td><td colspan="16">Application data Rate HRD</td></tr>
<tr><td>9</td><td colspan="16">SW Version</td></tr>
<tr><td>10</td><td colspan="16">Spare 1</td></tr>
<tr><td rowspan="3">App. Data</td><td>N+1</td><td colspan="16"></td></tr>
<tr><td></td><td colspan="16"></td></tr>
<tr><td>N+M</td><td colspan="16"></td></tr>
</table>

Table 5-5: Application Data transfer frame SM to Carrier DMS

5.5.1 Previous packet acknowledge

This field provides information of the receipt status of the previous package that has been sent to this individual Science Module. Broadcast messages are not acknowledged.

Values are:

- 0x0015 for NACK, data not received properly

© 2010 OHB System AG

- 0x0006 for ACK, data received properly
- 0x0018 for Transfer acknowledge, but Application Data not acknowledged

The Carrier DMS will retry transmission of the previous application data if this field is not set to ACK. However, the Carrier DMS runs maximum three attempts.

© 2010 OHB System AG

5.5.2 Type of Data

This field provides information of the application data type of the last package: Values are:

- 0x00 for No Data
- 0x54 for time
- 0x4D for messages
- 0x53 for telemetry package

5.5.3 SM Timer Status

The timer status shall be coded in one octet where each bit identifies a specific status value according to the definition given before. Sec. 3.8 provides details on the coding.

5.5.4 SM Mode

This field contains the current operational mode of the SM. Sec. 3.7 provides details on the coding.

5.5.5 Number of SM Messages

This field contains the number of SM messages ready for transfer on next request. The current messages ready that will be transferred by the current response package do not contribute.

The field is evaluated by the Carrier DMS. The Carrier DMS reduces the request cycle if this value is unequal to 0.

5.5.6 Number of SM TM Packets

This field contains the number of SM TM packages ready for transfer. The current TM packet that will be transferred by the current response package does not contribute.

The field is evaluated by the Carrier DMS. The Carrier DMS reduces the request cycle if this value is unequal to 0. However, SMs that have messages to be sent have higher priority.

© 2010 OHB System AG

5.5.7 DELETED

5.5.8 SM HK Acquisition Rate

The field contains the bulk HK generation cycle in seconds. Values can range from 2 to 30 seconds.

5.5.9 Current SM Process ID

The process ID shall be coded in a word.

5.5.10 Application Data Rate LAN

The LAN Net Application Data Rate shall be coded in one word. It identifies the net application data transfer over the EPM LAN. The value shall give the net rate in kbit/s measured over the last 5 seconds.

The ALIVE packets are included in this data rate.

5.5.11 Application Data Rate HRD

The HRD Net Application Data Rate shall be coded in one word. It identifies the net application data transfer over HRD.

The value shall give the net rate in kbit/s measured over the last 5 seconds.

5.5.12 SW Version

The software version shall be coded in one word.

5.5.13 DELETED

5.5.14 Application Data

The fields are not present if No data is transmitted. The following rules are applicable for different data types:

- Messages: maximum three messages can be transferred
- Telemetry packets, only one telemetry packet can be transferred
- Time, the 32 bit UTC coarse time will be transmitted

Application data if available shall be selected for transmission according to the following priority rules:

1. Time Packet
2. Messages
3. Telemetry (typically HK-report or selftest report over this interface)

© 2010 OHB System AG

5.5.14.1 Data Structure For Application Data

5.5.14.1.1 Time (Type 0x54)

The Time format is given in Table 5-6

Param #	Type	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
1	LU	1	EPM Coarse Time(High Word (#1))															
		2	EPM Coarse Time(Low Word (#2))															

Table 5-6: Time structure

Note: The time packet generation is only initialized upon the Set Subsystem Time command.

© 2010 OHB System AG

5.5.14.1.2 Messages (Type 0x4D)

Table 5-7 describes the application data structure for message transfer. A 32 bit unsigned integer gives the number of messages preceding the messages.

Param #	Type	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
1	LU	1	Number of Messages (Word #1)															
		2	Number of Messages (Word #2)															
		3	Message # 1															
		4																
		5																
		6																
		7																
		8																
		9																
		10																
		11																
		12																
		13																
		14																
		15																
		16																
		17	Additional messages, Maximum two further messages															
		Max 44																

Table 5-7: Application Data Structure for message transfer

The Message structure is defined in section 7.

5.5.14.1.3 Telemetry (Type 0x53)

The Telemetry structure is defined in section 7.

The destination field bits for FCC and LTU shall be set for all housekeeping TM packets (TM packet ID 0x0301) that are generated automatically and periodically by the Slave. Thus the value of this field for the HK-data TM transferred over RS-485 is 01010000/b. The SMSC routes the packets to the FCC for logging and downlink via Columbus LAN and to the LTU for display and further processing.

For all TM packets that are acquired by command the destination filed shall be set according to the command source field of the telecommand.

6. USB INTERFACE

The USB 1.1 (AD 5) standard is used.

The TCP/IP protocol as specified in section 4 is used. This is applicable to the on-board connection between SMSC and SM and the EGSE connection between Carrier EGSE and Science Module EGSE and the on-board connection between VU and LTU.

© 2010 OHB System AG

7. APPLICATION DATA TYPES

7.1 EPM Application Layer Telecommand Structure

The EPM internal telecommand structure is given in Table 7-1.

<table>
<tr><th>Field ID</th><th>Bit # / Word #</th><th>MSB</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>LSB</th></tr>
<tr><td rowspan="12">EPM Priv. Head.</td><td>1</td><td colspan="16">EPM Sync. Marker (Word #1 0xFFDB)</td></tr>
<tr><td>2</td><td colspan="16">EPM Sync. Marker (Word #2 0x5443)</td></tr>
<tr><td>3</td><td colspan="8">Command Source</td><td colspan="8">EPM Subsystem ID</td></tr>
<tr><td>4</td><td>EPM Time-tag Flag</td><td>Cmd Rep. ExecFlag</td><td colspan="2">EPM ChkSum Ind</td><td colspan="2">EPM TC Sequence Flag</td><td colspan="2">EPM Proc. Sequence Flag</td><td colspan="8">EPM Subsystem Unit ID</td></tr>
<tr><td>5</td><td colspan="16">EPM TC Sequence Counter</td></tr>
<tr><td>6</td><td colspan="16">EPM Procedure ID</td></tr>
<tr><td>7</td><td colspan="16">EPM Procedure Sequence Counter</td></tr>
<tr><td>8</td><td colspan="16">EPM Command ID</td></tr>
<tr><td>9</td><td colspan="14">Spare</td><td colspan="2">Command Priority</td></tr>
<tr><td>10</td><td colspan="16">EPM Number of words</td></tr>
<tr><td>11</td><td colspan="16">EPM Coarse Time (High Word (#1))</td></tr>
<tr><td>12</td><td colspan="16">EPM Coarse Time (Low Word (#2))</td></tr>
<tr><td rowspan="4">EPM User Words</td><td>13</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>14</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>N-2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>EPM Check Sum</td><td>N-1</td><td colspan="16">EPM Check Word</td></tr>
</table>

Table 7-1: EPM Telecommand Format

SM’s do not process the white fields.

The definition of the fields is:

7.1.1 EPM sync. marker

The synchronization marker is a double word. It is set to:

Byte 0 1 2 3

Hex	FF	DB	54	43
Bin	11111111	11011011	01010100	01000011

It can be used for internal transfer from SM unit to SM unit, e.g. if no further protocol support is foreseen.

TC and TM synchronization marker are different in order to allow determination of the structure by any application.

7.1.2 Command source

The command source specifies from where the telecommand was sent.

Telecommands can be part of a sequence, e.g. predefined procedures.

Coding is according the definition outlined in section 3.10.

7.1.3 EPM subsystem ID

The EPM subsystem ID specifies the sub-system of EPM or a SM to which the command is addressed.

Coding is according the definition outlined in section 3.6.

7.1.4 EPM time-tag flag

The time tag flag is used for the EPM command scheduler. If the flag is set to 1/b the command is time tagged. SM's use this field to identify whether a command is allowed to be executed as time tagged command.

It is SM responsibility to check the command accordingly.

7.1.5 Command Execution Report Generation Flag

The Command Execution Report Generation Flag regulates whether a command execution report is generated or not. If the flag is set to 1/b the report is generated.

Setting this flag is mandatory except for commands that are sent from the LTU for adjustments in Science Modules.

7.1.6 EPM check sum indicator

The EPM checksum indicator regulates how the receiver shall calculate the checksum.

Coding is according to the definition outlined in section 3.9.

7.1.7 EPM TC sequence flag

The EPM TC sequence flag is used to control the reception of complete command sequences by the Carrier DMS.

© 2010 OHB System AG

The sequence control is used for commands from ground only. The Carrier DMS evaluates telecommand sequences. The sequences can include commands for different addressees (e.g. a script loaded from ground). Thus SM does not use the sequence protocol layer.

Coding is according to the definition outlined in section 3.12.

7.1.8 EPM Procedure Sequence Flag

The Procedure Sequence Flag is used to control the receipt of command sequences from different sources. Different procedures can be started on these sources. Procedure sequence control will be performed for each procedure/source. The Carrier DMS is the only source that may run different procedures at the same time.

The procedures can include commands for different addressees. Control is performed by the Carrier DMS on the basis of command execution messages. Thus SM does not use the procedure protocol layer.

Coding is according to the definition outlined in section 3.12.

7.1.9 EPM subsystem unit ID

The EPM subsystem unit ID specifies the unit of a sub-system of EPM to which the command is addressed.

If the subsystem ID is set to ALL this field should be set 0.

7.1.10 EPM TC sequence counter

This counter is used for unique identification of commands from different sources. The counter is furthermore used to ensure the receipt of complete command sequences if the EPM TC sequence flag is set to 00/b (Continuation). The maximum number of telecommands in a sequence is limited to 100.

The sequence control is used for commands from ground only. The Carrier DMS evaluates telecommand sequences. The sequences can include commands for different addressees. Thus SM does not use the sequence protocol layer.

SM use this field only for filling the command execution message.

7.1.11 EPM procedure ID

The EPM Procedure ID identifies the current procedure form the source. The value 0x0 is reserved for manual commands or command sequences.

SM use this field only for filling the command execution message.

7.1.12 EPM procedure sequence counter

This Procedure Sequence Counter is incremented by one for each telecommand of a procedure. It is used to control that every command of a procedure is received.

© 2010 OHB System AG

The procedures can include commands for different addressees. Control is performed by the Carrier DMS on the basis of command execution messages. Thus SM does not use the procedure protocol layer.

SM use this field only for filling the command execution message.

7.1.13 EPM command ID

The EPM Command ID specifies the unique ID of a command for a given sub-system of EPM.

7.1.14 EPM command priority

This field specifies the priority of command execution.

Coding is according the definition outlined in section 3.11.

7.1.15 EPM number of words

The number of words is calculated over the entire EPM TC structure including the EPM check-word. The maximum value is 250, corresponding to 500 octets. For ground commands that are not transferred as data load packets, the size is limited to 40, corresponding to 80 octets.

The number of words for a TC without any user words (e.g. a Mode Change Command) is 13 (12 words header and one word checksum).

7.1.16 EPM coarse time

This double word gives the command execution time (UTC) in elapsed seconds from a defined epoch. The defined epoch is GPS time, i.e. midnight 5-6 January 1980.

If the command is not time tagged this field shall be set to 0.

In normal operation SM does not use this field. It can be used for test purposes e.g. to verify the overall execution time of urgent priority.

7.1.17 EPM user words

These fields can be used for command parameter. EPM provides a parameter format language, so that any non-standard protocol can also be supported by EPM.

7.1.18 EPM check sum

This field contains the check sum over the EPM private header and the EPM user data if the EPM Check sum indicator is set or at least a fixed bit pattern if the indicator is not set.

© 2010 OHB System AG

7.2 EPM Application Layer Telemetry Structure

The telemetry format is shown in Table 7-2.

<table>
<tr><th>Field ID</th><th>Bit # / Word #</th><th>MSB</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>LSB</th></tr>
<tr><td rowspan="15">EPM Priv. Head.</td><td>1</td><td colspan="16">EPM Sync. Marker (Word #1 0xFFDB)</td></tr>
<tr><td>2</td><td colspan="16">EPM Sync. Marker (Word #2 0x544D)</td></tr>
<tr><td>3</td><td colspan="8">EPM Subsystem Mode</td><td colspan="8">EPM Subsystem ID</td></tr>
<tr><td>4</td><td colspan="8">Destination</td><td colspan="8">EPM Subsystem Unit ID</td></tr>
<tr><td>5</td><td colspan="16">TM Identifier</td></tr>
<tr><td>6</td><td colspan="16">EPM Subsystem TM Counter</td></tr>
<tr><td>7</td><td colspan="8">Model</td><td colspan="8">SW Task ID</td></tr>
<tr><td>8</td><td colspan="16">EPM Subsystem Unit SW Version</td></tr>
<tr><td>9</td><td colspan="16">EPM Coarse Time (High Word (#1))</td></tr>
<tr><td>10</td><td colspan="16">EPM Coarse Time (Low Word (#2))</td></tr>
<tr><td>11</td><td colspan="16">EPM Fine Time</td></tr>
<tr><td>12</td><td colspan="8">EPM Subsystem Timer Status</td><td colspan="8">EPM Subsystem Experiment Mode</td></tr>
<tr><td>13</td><td colspan="14">Spare_1</td><td colspan="2">EPM ChkSum Ind</td></tr>
<tr><td>14</td><td colspan="8">Receiver Subsystem ID</td><td colspan="8">Receiver Subsystem Unit ID</td></tr>
<tr><td>15</td><td colspan="16">EPM Number of words</td></tr>
<tr><td rowspan="4">EPM User Words</td><td>16</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>17</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>N-2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>EPM Check Sum</td><td>N-1</td><td colspan="16">EPM Check Word</td></tr>
</table>

Table 7-2: EPM Telemetry Format

7.2.1 EPM Sync. Marker

The synchronization marker is a double word. It is set to:

Byte	0	1	2	3
Hex	FF	DB	54	4D
Bin	11111111	11011011	01010100	01001101

TC and TM synchronization marker are different in order to allow determination of the structure by any application.

© 2010 OHB System AG

7.2.2 EPM Subsystem Mode

This field contains the current operational mode of the Subsystem.

Coding is according the definition outlined in section 3.7.

7.2.3 EPM Subsystem ID

The EPM subsystem ID specifies the sub-system of EPM or a SM that has generated the packet.

Coding is according the definition outlined in section 3.6.

7.2.4 Destination

This field specifies where the packet shall be sent.

Coding is according the definition outlined in section 3.13.

7.2.5 EPM Subsystem Unit ID

The EPM subsystem unit ID specifies the unit of a sub-system of EPM that has generated the packet.

7.2.6 TM Identifier

This field specifies the unique ID for a Subsystem Unit of a source packet for a given Subsystem ID.

7.2.7 Subsystem TM Counter

The SM maintains this counter. It is incremented by one for each source packet for the corresponding data link. It is used to control that any packet is received. The counter shall be maintained for each destination individually.

7.2.8 Model

The model specifies which EPM model was used to record the data.

Coding is according the definition outlined in section 3.5.

7.2.9 SW Task ID

This field specifies the unique ID of the SW task for a given Sub-system and Unit of this subsystem that generated the packet.

7.2.10 Software Version

The software version shall be coded in one word.

© 2010 OHB System AG

It is used to ensure that the ground software, SM-MMI and the SM software are always consistent.

Coding is according the definition outlined in section 3.15.

7.2.11 EPM Coarse Time

This double word gives the telemetry generation time (UTC) in elapsed seconds from a defined epoch.

The defined epoch is GPS time, i.e. midnight 5-6 January 1980.

The time shall be set once all data for a telemetry packet are available by the originator.

7.2.12 EPM Fine Time

This word shall give the fine time with a precision of 0.1 ms.

The maximum value is 9,999 ("Unit 0.1 ms").

7.2.13 Subsystem Timer Status

The timer status shall be coded in one octet where each bit identifies a specific status value.

Coding is according the definition outlined in section 3.8.

7.2.14 EPM Subsystem Experiment mode

The experiment mode of the SM is coded in this field.

7.2.15 Spare_1

This field is not used. It is set to zero.

7.2.16 DELETED

7.2.17 EPM Check Sum Indicator

The checksum indicator specifies how to calculate the check-sum. Coding is according to the definition outlined in section 3.9.

7.2.18 Receiver Subsystem ID

The EPM subsystem ID of the receiver of the TM packet is used if the destination is set to INTERNAL. For all other destinations the field is 0.

7.2.19 Receiver Subsystem UNIT ID

The EPM subsystem unit ID of the receiver of the TM packet is used if the destination is set to INTERNAL. For all other destinations the field is 0.

© 2010 OHB System AG

7.2.20 EPM number of words

The number of valid user words is calculated over the entire EPM TM structure including the EPM check-word. The maximum value is 700 words, corresponding to 1400 octets.

The number of words for a TM without any user words (e.g. Time Status Report) is 16 (15 words header and one word checksum).

7.2.21 EPM User Words

These fields contain the application data.

7.2.22 EPM Check Sum

This field contains the check sum over the EPM private header and the EPM user data if the EPM Check sum indicator is set or at least a fixed bit pattern if the indicator is not set.

Coding is according the definition outlined in section 3.9.

© 2010 OHB System AG

7.3 EPM Message Structure

The message format is shown in Table 7-3.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
EPM Msg. Head.	1	Message Type								EPM Subsystem-ID							
	2	Message Identifier															
	3	EPM Subsystem Unit Message Counter															
	4	EPM Subsystem Unit ID								SW Task ID							
	5	EPM Coarse Time (High Word (#1))															
	6	EPM Coarse Time (Low Word (#2))															
	7	EPM Fine Time															
	8	Spare 1															
EPM Msg Data	9																
	10																
	11																
	12																
	13																
	14																

Table 7-3: EPM Message Format

7.3.1 Message Type

Message types are used to identify the type of the message. Values are:

- 1 for Information message
- 2 for warning message
- 3 for error message
- 4 for configuration change message
- 5 for mode change message
- 6 for command execution/rejection message

7.3.2 EPM Subsystem ID

The EPM Subsystem-ID specifies the sub-system of EPM or a SM from which the message is generated.

7.3.3 Message Identifier

This field specifies the unique ID of a message for a given sub-system and Unit of this sub-system.

© 2010 OHB System AG

7.3.4 Subsystem Unit Message Counter

The SM maintains this counter. It is incremented by one for each message. It is used to control that any message is received.

7.3.5 EPM Subsystem Unit ID

The EPM subsystem unit ID specifies the unit of a sub-system of EPM from which the message is generated.

7.3.6 SW Task ID

This field specifies the unique ID of the SW task for a given sub-system and Unit of this subsystem that generated the message.

7.3.7 EPM Coarse Time

This double word gives the command execution time (UTC) in elapsed seconds form a defined epoch.

The defined epoch is GPS time, i.e. midnight 5-6 January 1980.

7.3.8 EPM Fine Time

This word shall give the fine time as counter of 0.1 ms.

7.3.9 EPM Message Data

The EPM Message Data contain additional information. If not all bytes are used then the unused bytes shall be set to 0x16.

© 2010 OHB System AG

7.4 Ancillary Data

The environmental data taken from the Columbus ancillary data are:

1. Average cabin temperature (8bit)
2. Relative humidity in air cabin (12bit)
3. Partial Pressure of Oxygen in the cabin (12bit)
4. Partial pressure of Carbon Dioxide in the cabin (12bit)
5. Total pressure in the cabin (12bit)
6. Ku-Band and S-Band Status (8bit)

Table 7-4 shows the data structure for the Columbus provided ancillary data.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Col Anc. Data	1	Air Humidity 1															
	2	Air Humidity 2															
	3	Partial Pressure O_2 Sensor 1															
	4	Partial Pressure O_2 Sensor 2															
	5	Partial Pressure CO_2 Sensor 1															
	6	Partial Pressure CO_2 Sensor 2															
	7	Total Pressure 1															
	8	Total Pressure 2															
	9	Total Pressure 3															
	10	Total Pressure 4															
	11	Cabin Temperature 2								Cabin Temperature 1							
	12	Ku-Band Status								S-Band Status							

Table 7-4: EPM Ancillary Data

The accuracy and point pair calibration of the ECLS sensors is described in the following sections.

7.4.1 Relative humidity in the cabin

Accuracy: ± 2.7 % full scale respectively: ± 2.7 % rH
(nominal operational range: 25 % rH to 70 % rH)

Conversion Routine:

© 2010 OHB System AG

Raw Value	Engineering Value
0	0 pctrH
4095	100 pctrH

Table 7-5: Relative Humidity Point Pair Calibration

Unused upper 4 bits must be set to zero.

7.4.2 Partial oxygen pressure in the cabin

Conversion Routine:

Raw Value	Engineering Value
0	-5,2764 mmHg
2001	-5,1666 mmHg
4095	225,02 mmHg

Table 7-6: Partial Oxygen Pressure Point Pair Calibration

Unused upper 4 bits must be set to zero.

7.4.3 Partial CO2 pressure in the cabin

Conversion Routine:

Raw Value	Engineering Value
0	-0,68914 mmHg
2001	-0,68914 mmHg
4095	15,316 mmHg

Table 7-7: Partial Carbon Dioxide Pressure Point Pair Calibration

Unused upper 4 bits must be set to zero.

7.4.4 Total pressure in the cabin

Conversion Routine:

Raw Value	Engineering Value
0	-19,347 mmHg
2001	-18,944 mmHg
4095	825,07 mmHg

Table 7-8: Total Pressure Point Pair Calibration

Unused upper 4 bits must be set to zero.

© 2010 OHB System AG

7.4.5 Average Cabin Temperature

Accuracy: = ± 1.42 degC full scale (nominal operational range: +10 degC to + 40 degC)

Conversion Routine:

Raw Value	Engineering Value
0	-25 degC
100	10 degC
134	22 degC
156	30 degC
185	40 degC
255	65 degC

Table 7-9: Average Cabin Temperature Point Pair Calibration

7.4.6 The KU-Band and S-Band

The AOS/LOS downlink status decoding is:

- 1 for AOS
- 0 for LOS

© 2010 OHB System AG

8. SM-MMI CONTROL DATA TYPES

8.1 Directives

SM-MMI Directives are used to control the SM-MMI. The directives follow the general layout as shown in Table 8-1.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Dir. Head.	1	Directive-ID								Software Unit ID of MMI							
	2	Number of words															
Dir. Data	3																
	N-2																
	N-1																

Table 8-1: EPM Directive Format

The software unit ID of MMI is defined in Table 4-2. This is the ID of the receiver of the instruction. The Directive ID defines the structure if the directive data that follows the directive header.

The number of words is always calculated over the entire data structure.

Table 8-2 shows the relation between the instructions of IAP as defined in RD1 and the directives belonging to packet type DIRECTIVES (0xBB44).

IAP Instruction	Directive	Directive ID
Start MMI	Set Environment	1
	Get User Profile	5
OpenMMIDisplay	Show Dialog	2
CloseMMIDisplay	Close Dialog	3
StopMMI	Shutdown MMI	4
ShowMMI	Set MMI Visibility	6

Table 8-2: Relation between IAP instructions and directives

Directives must be acknowledged after at least 3 seconds.

© 2010 OHB System AG

8.1.1 Set Environment

The Set Environment Directive defines the SM-MMI runtime environment.

This directive is sent to the MMI during the MMI startup procedure. The difference for on board dialogs if called by IAP or manually by the CLSW Manual Access Display Menu is the usable y-range for the MMI. A MMI may provide more features if called in an EGSE environment. E.g. the CLSW manual commanding menu for the EGSE supports time tagging and command sequencing in addition to the features provided by the onboard dialog.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Dir. Head.	1	Directive-ID = 1								Software Unit ID of MMI							
	2	Number of words = 4															
Dir. Data	3	Environment ID															
	4	EPM Dialog ID															

Table 8-3: EPM Set Environment Directive Format

The following values are valid for the Environment ID:

- 0x0050 if called and controlled by IAP
- 0x004D if called manually by CLSW
- 0x0045 if called in an EGSE environment

The EPM Dialog ID is the ID of the selected dialog at SM-MMI program start.

© 2010 OHB System AG

8.1.2 Show Dialog

The show dialog directive forces the SM-MMI to display the selected dialog at given position and size. User specific data can be submitted in addition.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Dir. Head.	1	Directive-ID = 2								Software Unit ID of MMI							
	2	Number of words															
Show Dialog Head.	3	X-Position															
	4	Y-Position															
	5	X-Length															
	6	Y-Length															
	7	EPM Dialog ID															
	8	Spare = 0															
User Data	9																
	N-2																
	N-1																

Table 8-4: EPM Show Dialog Format

Positions and length for the selected EPM Dialog ID are given in number of pixel. Details are described in RD1.

The User Data field can be used optionally to transfer MMI specific data if required. User data must be aligned to double words. If fill bytes are required these bytes are set to 0x16.

8.1.3 Close Dialog

The Close dialog directive closes a dialog of the addresses MMI.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Dir. Head.	1	Directive-ID = 3								Software Unit ID of MMI							
	2	Number of words = 4															
Dir. Data	3	EPM Dialog ID															
	4	Spare = 0															

Table 8-5: EPM Close Dialog Directive Format

© 2010 OHB System AG

8.1.4 Shutdown MMI

The shutdown directive forces a MMI to disconnect the TCP/IP connection and to close the application.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Dir. Head.	1	Directive-ID = 4								Software Unit ID of MMI							
	2	Number of words = 2															

Table 8-6: EPM Shutdown SM-MMI Directive Format

8.1.5 Get User Profile

The user profile interface is a link to a user data file that comprises at least the following information:

- Crew procedures for this user, model and increment
- Relevant physiological data
- List of Displays and user defined display size and position

The further content of the file depends on the mission increment and the experiments selected.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Dir. Head.	1	Directive-ID = 5								Software Unit ID of MMI							
	2	Number of words															
User Profile File Name	3	File Name (including path) (NULL terminated with fill bytes)															
	N-2																
	N-1																

Table 8-7: EPM Get User Profile Format

The filename is a string terminated with 0x00 (\0). If this null terminated string does not align to double words, the remaining bytes are set to 0x16.

8.1.6 Set MMI Visibility

The Set MMI visibility directive to bring the MMI to foreground or to hide it

© 2010 OHB System AG

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Dir. Head.	1	Directive-ID = 6								Software Unit ID of MMI							
	2	Number of words = 4															
Dir. Data	3	MMI Visibility															
	4	Spare = 0															

Table 8-8: EPM Set MMI Visibility Directive Format

The MMI Visibility values are:

- 0x0000 for hide MMI
- 0x00FF for show MMI

8.2 Directive Acknowledge

The directive acknowledges provides feedback on the directive execution.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Ack.	1	Directive ID								Software Unit ID of MMI							
	2	Acknowledge Status															

Table 8-9: EPM Directive Acknowledge Format

The directive ID field carries the directive that is acknowledged.

Acknowledge status is:

- 0x00 if successfully executed
- 0xFF in case of failure

© 2010 OHB System AG

8.3 Setting Instruction

The SM-MMI may have a setting or tuning menu (synoptic displays).

These settings are passed to the IAP. The IAP generates commands from this information and issues them to the SM. Thus the IAP inserts the procedure sequence counter. The settings data structure contains thus:

- ID of the EPM dialog that issues the instruction
- Number of words that are transferred
- Name of the addressed SM
- Name of the addressed SM unit
- Name of the command
- Regulation field that specifies, whether a command execution report shall be generated or not.
- Checksum indication (see Section 3.9)
- Parameter String for the command (as used in the IAP environment)

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Set. Head.	1	EPM Dialog ID															
	2	Number of words															
	3	EPM Subsystem-ID								EPM Subsystem Unit ID							
	4	EPM Command ID															
	5	Cmd. Ex Rep. Flag								Checksum Indicator							
	6	Spare =0															
Set. String	7																
	N-2																
	N-1																

Table 8-10: EPM Setting Instruction Format

The parameter string has to be NULL terminated and filled up with 0x16 to ensure double word alignment of the structure.

© 2010 OHB System AG

8.4 Procedure Message

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Poc. Msg. Head.	1	EPM Dialog ID															
	2	Number of words															
Proc. Msg. Text	3																
	N-2																
	N-1																

Table 8-11: EPM Procedure Message Format

The EPM Dialog ID is the ID of the sender. The procedure message text has to be NULL terminated and filled up with 0x16 to ensure double word alignment of the structure.

Procedure messages are displayed in a standard dialog of the IAP environment.

© 2010 OHB System AG

9. FILE TRANSFER

File transfer can take place over LAN (USB or Ethernet) or as downlink via the HRD interface of the SMSC.

Files are transferred as stream of TM packets internally and to ground. The receipt of a packet is not acknowledged individually. These packets have an internal secondary header that supports a sequence layer as for telecommands (sequences start, continue and end with increment of the sequence counter for each packet). This allows checking for complete receipt of all packets by the receiver.

Only one file transfer can take place at the time for each Software Unit. The time interval between two file transfer initializations must be at least 2 seconds.

The first packet contains the full source and destination-filenames including path of the file and other file attributes. The coarse time of the start of transfer is included in each packet to allow unique identification of the packet to a file. The combination of the fields: sender subsystem, sender subsystem unit and coarse start time is used to identify the file transfer action unambiguously.

Files are not deleted automatically after downlink on board. If a file is not received completely on ground, the FRC can reinitiate the file transfer by command. (Packets from both transfer actions may be combined to get a complete file).

The destination field of the TM packets is set to INTERNAL for internal file transfer. The receiver subsystem field and the receiver subsystem unit field specify the receiver of the file.

The sender and the receiver have to be operated in setup mode for file transfer over LAN. There shall be no Science Data traffic during file transfer.

The time between sending of two packets shall be adjusted according to the data rate budget of the interface used. The minimum time of silence between two packets is 1ms.

The software items related to file transfer are standardized. Details are described in AD3.

Max File Size to be transferred by EPM File Transfer is 50 or 84,5 MByte,
for SMSC & VU: 1352 octets x 65.534 Packet Segments = 88601968 Bytes = 84,5 MByte
for FCC: 800 octets x 65.534 Packet Segments = 52427200 Bytes = 50MByte

9.1 File Name Length Conventions

The maximum length of pathnames is 200 octets.

The filename length is if coded in TC parameter (no terminator) is:

- for a DOS-File System 12 (8.3)
- for WIN32 systems: maximal 31

The filename length is if coded in TM packets (NULL terminator included) is:

- maximal 32 (Long File Name Space: file names + string terminator (\0) + fill bytes)

© 2010 OHB System AG

The individual drives, type of operating system and used separator in directory paths of the subsystem units is defined in the unique ICD’s.

© 2010 OHB System AG

9.2 File Management Telecommands

All telecommands belonging to file management have the command group identifier 0x5.

Table 9-1 provides an overview on the file management commands (for a DOS file system, filename length is limited to 212).

Command	Command ID	Command Parameter String (DOS file system)
Down Link File	0x501	uL(0x0002,0x0004,0x0006)sS(1,212)sS(1,231)
Send File	0x502	usS(1,212)sS(1,231)
Copy File	0x503	sS(1,212)sS(1,212)
Delete File	0x504	sS(1,212)
Rename File/Directory	0x505	sS(1,212)sS(1,212)
List Directory	0x506	-
Change Directory	0x507	sS(1,200)
Make Directory	0x508	sS(1,200)
Remove Directory	0x509	sS(1,200)
Cancel File Transfer	0x50A	lu

Table 9-1: File management commands overview

All commands belong to internal file transfer except the DOWNLINK File command. The copy file command is applicable to file copies on the same unit.

Note: The format instruction s denotes “alphanumeric”. This is a structure consisting of a 16bit unsigned and a string (no terminator). The 16bit unsigned gives the length of the succeeding string in bytes.

9.2.1 Downlink File

The addressed unit generates a File TM stream with destination field set to FRC, USOC or USOC & FRC upon receipt of a DOWNLINK file command. The receiver subsystem and receiver subsystem ID are set to 0. The command parameter string for DOS-File system is: uL(0x0002,0x0004,0x0006)sS(1,212)sS(1,231).

The parameters are:

1. unsigned (u): Destination of the telemetry, FRC, USOC, or FRC and USOC
2. alphanumeric (s): Full Filename of the sender
3. alphanumeric (s): Full Filename of the receiver

9.2.2 Send File

The addressed unit generates a File TM stream with destination field set to INTERNAL upon receipt of a Send File command. The receiver subsystem and receiver subsystem ID are set

© 2010 OHB System AG

according to the first parameter of the command. The command parameter string for DOS-File system is: usS(1,212)sS(1,231).

The parameters are:

1. unsigned (u): MSB Byte identifies SUBSYSTEM of the Receiver, LS Byte identifies receiver subsystem unit
2. alphanumeric (s): Full Filename of the sender
3. alphanumeric (s): Full Filename of the receiver

9.2.3 Copy File

The Copy File command copies a file locally.

The command parameter string for DOS-File system is: sS(1,212)sS(1,212).

The parameters are:

1. alphanumeric (s): Full Filename of the file to be copied
2. alphanumeric (s): Full Filename of the target file

9.2.4 Delete File

The Delete File command deletes a file. The command parameter string for DOS-File system is: sS(1,212).

The parameter is the name of the file to be deleted.

9.2.5 Rename File/Directory

The Rename File/Directory command renames a file or directory locally.

The command parameter string for DOS-File system is: sS(1,212)sS(1,212).

The parameters are:

1. alphanumeric (s): Old Full Filename or directory name
2. alphanumeric (s): New Full Filename or directory name

9.2.6 List Directory

The List Directory Command generates Directory Listing TM packets.

9.2.7 Change Directory

The Change Directory command changes the working directory. The command parameter string is: sS(1,200).

© 2010 OHB System AG

The parameter is the name of the directory.

9.2.8 Make Directory

The Make Directory command generates a new directory. The command parameter string is: sS(1,200).

The parameter is the name of the directory.

9.2.9 Remove Directory

The Remove Directory command deletes the selected directory. The command parameter string is: sS(1,200).

The parameter is the name of the directory.

9.2.10 Cancel File Transfer

The CANCEL File Transfer command stops a file transfer. The command parameter string is: lu.

The parameter is a long unsigned. It is the coarse start time of the file transfer to be cancelled.

9.3 File Management Telemetry

File Management telemetry has the group identifier 0x04. Two different packets are defined:

1. Directory listing (ID 0x0402)
2. File Packets (ID 0x0401)

The directory listing TM are generated upon receipt of the List Directory Command. File Packets are generated upon receipt of the Downlink File Command or the Send File Command.

These commands start a process that generates a sequence of packets. Therefore a secondary header is introduced that covers the sequence management and the unique identification of the sequence.

© 2010 OHB System AG

9.3.1 Secondary Header Sequence Control

The header contains a sequence control area. Table 9-2 shows the structure of the secondary header sequence control.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Seq. Ctrl.	1	Spare = 0														Seq. Flag	
	2	Secondary Header Sequence Counter															

Table 9-2: Secondary Header Sequence Control Structure

The secondary header sequence flags are coded according to the rules provided in Table 3-9. The LSB is the start bit and the MSB is the stop bit. For standalone packets both bits are set.

The secondary header sequence counter of the first packet of a sequence is 0. The sequence counter is incremented by one for each packet.

9.3.2 Directroy listing TM

The TM ID is 0x0402.

Table 9-3 shows the complete secondary header.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Seq. Ctrl.	1	Spare = 0														Seq. Flag	
	2	Secondary Header Sequence Counter															
Curr. Dir.	3	Current Working Directory Name (NULL terminated with fill bytes)															
	102																
	103																
	104	Spare 0x1616															

Table 9-3: Directory Listing TM Secondary Header

The current directory name is given as a NULL terminated string. Unused bytes are set to 0x16. The filed length is 202 octets.

The header is succeeded by Directory Item Structures. A Directory Listing TM-packet can contain a maximum of 24 Directory Item Structures.

If the directory listing has more entries then further packets are generated and the sequence control fields are set accordingly.

Table 9-4 describes the Directory Item Structure.

© 2010 OHB System AG

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Dir. Item	1	Item Name (NULL terminated with fill bytes)															
	2																
	..																
	..																
	..																
	16																
	17	Item Size															
	18																
	19	Last Change Coarse Time															
	20																
	21	Item Attribute															
	22																

Table 9-4: Directory Item Structure

The item name is a NULL-terminated string. Unused bytes are set to 0x16.

The item size is:

- for Files: the size in byte
- set to 0 for directories

The Last Change Coarse Time is the Coarse Time Value when the last change of the file/directory has taken place.

The Item Attribute specifies which kind of item is documented. Values are:

- 0x00000010 for directories
- 0x00000020 for files

9.3.3 File Packet TM

The TM-ID of File Packet TM is 0x0401.

File Packet TM include a secondary header as shown in Table 9-5, which is succeeded by data.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Seq. Ctrl.	1	Spare = 0														Seq. Flag	
	2	Secondary Header Sequence Counter															
	3	Length of file data															

© 2010 OHB System AG

File	4	Spare = 0
Trans.	5	File Size
Ctrl.	6	
Block	7	File Transfer Start Coarse Time
	8	

Table 9-5: File Packet TM Secondary Header

The File Transfer Control Block contains the length of valid file data that the packets contain. Unused bytes of data are set to 0x16.

The File Size is given in number of bytes. It is recommended to use the maximum TM length for file transfer. This is 1352 octet.

The File Transfer Start Coarse Time is used to uniquely identify the file transfer between sender and receiver. It is the time when the first packet is generated.

Files are always transferred as a sequence of packets. The first packet of a file transfers is the File Transfer Init Packet. Its secondary header sequence flag is set to 01/b and the secondary header sequence counter is 0.

The data of the File Transfer Init packet are the source file name and the destination file name. Both are put to string field s of 232 octet size. They are NULL-terminated strings. Unused bytes are set to 0x16.

The following packets contain the file content. Their sequence flag are all 0.

The last packet of the file transfer is the File Transfer Completion Packet. Its sequence flag is set to 10/b. It does not contain any data.

A file packet sequence consists of a minimum of three packets. The sequence is:

1. File Transfer Init Packet
2. File Transfer Data Packet(s)
3. File Transfer Completion Packet

The following section give an example of the packets for file transfer with MAX_NUM packets, a start time START_TIME and a file size FILE_SIZE.

9.3.3.1 File Transfer Init Packet Application Data Layout

The File Transfer Init Packet Application Data layout is:

© 2010 OHB System AG

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Seq. Ctrl.	1	0														01/b	
	2	0															
File Trans. Ctrl. Block	3	0															
	4	0															
	5	FILE_SIZE															
	6																
	7	START_TIME															
	8																
File Names	9	Source File Name (NULL terminated with fill bytes)															
	..																
	..																
	124																
	125	Destination File Name (NULL terminated with fill bytes)															
	..																
	..																
	240																

Table 9-6: File Transfer Init Packet Application Data Layout

9.3.3.2 File Transfer Data Packet Application Data Layout

The File Transfer Data Packet Application Data layout is:

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Seq. Ctrl.	1	0														00/b	
	2	0 < Sequence Count < MAX_NUM															
File Trans. Ctrl. Block	3	1 < Number of valid File Data octets < 1353															
	4	0															
	5	FILE_SIZE															
	6																
	7	START_TIME															
	8																
File Data	9	File Data															
	..																
	..																
	N																

Table 9-7: File Transfer Data Packet Application Data Layout

© 2010 OHB System AG

9.3.3.3 File Transfer Completion Packet Application Data Layout

File Transfer Completion Packet Application Data layout is:

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
Seq. Ctrl.	1	0														10/b	
	2	Sequence Count <= MAX_NUM															
File Trans. Ctrl. Block	3	0															
	4	0															
	5	FILE_SIZE															
	6																
	7	START_TIME															
	8																

Table 9-8: File Transfer Completion Application Data Packet Layout

9.4 File Management Messages

The sender shall generate a File Transfer Started Message upon start of file transfer. The receiver shall generate a Nominal File Transfer termination message after successful completion of storage of the file. Both messages have Message Type INFO (1). In case of a detected abort of a file transfer the File Transfer Abort message shall be generated as WARNING (Message Type =2). The sender (e.g. time out) or the receiver can generate this warning.

Table 9-9 shows the format for the File transfer Start/stop/abort/timeout message.

Field ID	Bit # / Word #	MSB	1	2	3	4	5	6	7	8	9	10	11	12	13	14	LSB
EPM Msg. Head.	1	Message Type=1 or 2								EPM Subsystem-ID							
	2	Message Identifier															
	3	EPM Subsystem Unit Message Counter															
	4	EPM Subsystem Unit ID								SW Task ID							
	5	EPM Coarse Time (High Word (#1))															
	6	EPM Coarse Time (Low Word (#2))															
	7	EPM Fine Time															
	8	Spare 1															
EPM Msg Data	9	Sender Subsystem-ID								Sender Subsystem Unit ID							
	10	Receiver Subsystem-ID								Receiver Subsystem Unit ID							
	11	File Transfer Start Coarse Time (from Sender)															
	12																
	13	0x1616															
	14	0x1616															

Table 9-9: EPM File Transfer Status Message

Table 9-10 provides a list of predefined File Transfer Status messages.

Message Identifier	Description	Name	Message Type	Display
0x0601	File Transfer/Copy started	File Transfer Start	1	CLSW Msg Display
0x0602	File Transfer/Copy terminated nominally	File Transfer. Stop	1	CLSW Msg Display
0x0603	File Transfer/copy aborted	File Transfer. Aborted	2	CLSW Msg Display
0x0604	File Receive timeout message	File Receive Timeout	2	CLSW Msg Display

Table 9-10: Predefined message identifier for File Transfer status

The message data contain the subsystem ID and unit ID of the sender and the receiver and the File Transfer Coarse Start Time of the sender as inserted in each secondary header of the TM packets of the file transfer. The messages are also applicable to internal file copies.

The timeout for receive timeout is 30 seconds.

© 2010 OHB System AG

10. ABBREVIATIONS AND ACRONYMS

ACS	Automatic Command Scripts
AD	Applicable Document
AD	Architectural Design
ADD	Architectural Design Document
ADP	Acceptance Data Package
AIT	Assembly, Integration and Test
AIV	Assembly, Integration and Verification
APM	Attached Pressurized Module
APID	Application Process Identifier
ASCII	American Standard Code for Information Interchange
ASW	Application SoftWare
BDC	Baseline Data Collection
BOB	Break-Out Box
BSW	Basic SoftWare
CAM	Commercial, Aviation, and Military (equipment)
CASE	Computer Aided Software Engineering
CARRIER DMS	Carrier Data Management System
CCB	Configuration Control Board
CCI	Crew Command Interface
CCSDS	Consultative Committee on Space Data Systems
CDR	Critical Design Review
CLSW	Carrier control Laptop SoftWare
CM	Configuration Management

© 2010 OHB System AG

COTS	Commercial Of-The-Shelf software
CPAH	Columbus Payload Accommodation Handbook
CSCI	Computer Software Configuration Item
CPU	Central Processing Unit
DaSS	Data Services System
DD	Detailed Design
DDD	Detailed Design Document
DMS	Data Management System
DRD	Documents Requirements Description
DRL	Document Requirement List
EAC-TM	European Astronaut Centre Training Model
EGSE	Electronic Ground Support Equipment
ESW	EPM Facility SoftWare
FDIR	Failure Detection, Isolation, and Recovery
FFPT	Full Function and Performance Test
FM	Flight Model
FRC	Facility Responsible Centre
FST	Facility Science Team
GCP	Guided Crew Procedure
GM	Ground Model
GSE	Ground Support Equipment

© 2010 OHB System AG

HK	HouseKeeping data
HRDL	High Rate Data Link
HRF	Human Research Facility
IAP	Interactive Automatic crew Procedure
ICD	Interface Control Document
IDE	Integrated Development Environment
I/F	InterFace
IMS	Inventory Management System
I/O	Input/Output
ISS	International Space Station
JSC	Johnson Space Center
KSC	Kennedy Space Center
LAN	Local Area Network
LOS	Loss Of Signal
LTU	LapTop computer Unit
Mbps	Mega bit Per Second
Mbyte	Mega byte
MI	Mission Increment
MIPS	Million Instructions Per Second
MMI	Man Machine Interface
MTL	Master TimeLine

© 2010 OHB System AG

NTP	Network Time Protocol
ODF	Operations Data File
OM	Operation and Maintenance
OOL	Out-Of-Limit
ORU	On-orbit Replaceable Unit
OS	Operating System
OSTP	Onboard Short-Term Plan
PA	Product Assurance
PC	Portable Computer (Laptop)
P/L	PayLoad
PLSS	PayLoad Software Specification
PUI	Programme Unique Identifier
RD	Reference Document
RTU	Remote Terminal Unit.
SCP	SCience Process
SCOE	Special Check-Out Equipment
SDE	Software Development Environment
SFPT	Short Function and Performance Test
SID	Software item IDentifier
SM	Science Module
SMIRD	Science Module Interface Requirements Document

© 2010 OHB System AG

SNTP	Simple Network Time Protocol
SPLC	Standard Payload Computer
SPOE	Standard Payload Outfitting Equipment
SR	Software Requirements
SRB	Software Review Board
SRD	System Requirements Document
S/S	Sub-System
SW	SoftWare
SwRD	SoftWare Requirements Document
SWRU	SoftWare Replaceable Unit
SWVCD	SoftWare Verification Control Document
TBC	To Be Confirmed
TBD	To Be Defined
TC	TeleCommand
TM	Telemetry
TM/TC	Telemetry and Telecommand
TR	Transition
UDP	Utility Distribution Panel
UHB	User Home Base
UR	User Requirements
USOC	User Support and Operations Center
UTC	Universal Time Correlated
XSM	Extended Science Module

© 2010 OHB System AG