

DIN EN 16603-50-04

ICS 49.140; 49.090; 33.040.40

**Raumfahrttechnik –
Telekommando-Protokolle, -synchronisation und -kanalkodierung;
Englische Fassung EN 16603-50-04:2014**

Space engineering –
Space data links –
Telecommand protocols, synchronization and channel coding;
English version EN 16603-50-04:2014

Ingénierie spatiale –
Liaisons de données spatiales –
Protocoles, synchronisation et codage canal des liaisons télécommande;
Version anglaise EN 16603-50-04:2014

Gesamtumfang 159 Seiten

DIN-Normenausschuss Luft- und Raumfahrt (NL)

Nationales Vorwort

Dieses Dokument (EN 16603-50-04:2014) wurde vom Technischen Komitee CEN/CLC/TC 5 „Raumfahrt“ erarbeitet, dessen Sekretariat vom DIN (Deutschland) gehalten wird.

Das zuständige deutsche Normungsgremium ist der Arbeitsausschuss NA 131-10-01 AA „Interoperabilität von Informations-, Kommunikations- und Navigationssystemen“ im DIN-Normenausschuss Luft- und Raumfahrt (NL).

Dieses Dokument (EN 16603-50-04:2014) basiert auf ECSS-E-ST-50-04C.

Dieses Dokument enthält unter Berücksichtigung des DIN-Präsidialbeschlusses 1/2004 nur die englische Originalfassung von EN 16603-50-04:2014.

Dieses Dokument wurde speziell zur Behandlung von Raumfahrtsystemen erarbeitet und hat daher Vorrang vor jeglicher Europäischer Norm, da es denselben Anwendungsbereich hat, jedoch über einen größeren Geltungsbereich (z. B. Luft- und Raumfahrt) verfügt.

Nationaler Anhang NA (informativ)

Begriffe und Abkürzungen

3 Begriffe und Abkürzungen

Begriffe aus anderen Normen

Für die Anwendung dieses Dokuments gelten die Begriffe nach ECSS-S-ST-00-01.

3.2 Für diese Norm spezifische Begriffe

3.2.1

Missionsphase

Zeitraum innerhalb einer Mission, in dem festgelegte fernsteuerungsbezogene Merkmale erreicht werden

Anmerkung 1 zum Begriff Der Übergang zwischen zwei aufeinanderfolgenden Missionsphasen kann zu einer Unterbrechung der Dienste der ferngesteuerten Datenübertragungsleitung führen.

3.2.2

Oktett

Gruppe aus acht Bits

Anmerkung 1 zum Begriff Die Benummerung für Oktette innerhalb einer Datenstruktur beginnt mit 0.

Anmerkung 2 zum Begriff Siehe 3.4 für die Benummerungskonvention für Bits.

3.2.3

Paket

Datenstruktur veränderlicher Länge, bestehend aus Nutzerdaten höherer Schicht, die in die genormten Vorlaufinformationen eingebunden sind

3.2.4

statisch

ohne Änderungen innerhalb eines spezifischen virtuellen Kanals

Anmerkung 1 zum Begriff Diese Norm enthält Anforderungen an die Unveränderlichkeit bestimmter Merkmale der darin festgelegten Datenstrukturen über eine oder sämtliche Missionsphase(n).

Anmerkung 2 zum Begriff Siehe 4.7 für den virtuellen Kanal.

3.2.5

Verfahren des Betriebs in der Bitübertragungsschicht

Verfahren in der Bitübertragungsschicht zum Aktivieren und Deaktivieren des physikalischen Kommunikationskanals durch Anwenden der HF-Trägerfrequenz- und -modulationsverfahren

3.3 Abkürzungen

Für die Anwendung dieses Dokuments gelten die Abkürzungen in ECSS-S-ST-00-01 und die folgenden Abkürzungen:

Abkürzung Bedeutung

AD	Abnahmeprüfung und -daten (en: acceptance check and data)
ARQ	automatische Wiederholungsanfrage (en: automatic repeat request)
BC	Umgehung der Abnahmeprüfung und -kontrolle (en: bypass of acceptance check and control)
BCH	Bose-Chaudhuri-Hocquenghem (en: Bose-Chaudhuri-Hocquenghem)
BD	Umgehung der Abnahmeprüfung und -daten (en: bypass of acceptance check and data)
BER	Bitfehlerquote (en: bit error rate)
BTG	Bitübertragungsgenerator (en: bit transition generator)
CCSDS	Beratender Ausschuss für Weltraumdatensysteme (en: Consultative Committee for Space Data Systems)
CLCW	Steuerbefehl der Kommunikationsverbindung (en: communications link control word)
CLTU	Übertragungseinheit der Kommunikationsverbindung (en: communications link transmission unit)
CMM	Modul zur Trägermodulation (en: carrier modulation mode)
COP	Kommunikations-Betriebsverfahren (en: communications operation procedure)
FARM	Rahmenabnahme- und Berichtsmechanismus (en: frame acceptance and reporting mechanism)
FDU	Dateneinheit des Fernsteuerungsrahmen (en: telecommand frame data unit)
FECF	Rahmenfehler-Kontrollfeld (en: Frame Error Control Field)
FOP	Rahmen-Betriebsverfahren (en: frame operation procedure)
ID	Identifikator oder Identifikation (en: identifier, or, identification)
LLIF	Schnittstelle an der unteren Schicht (en: lower layer interface)
MAP	Multiplexer-Zugangspunkt (en: multiplexer access point)
MSB	höchstwertiges Bit (en: most significant bit)
NRZ-M	Non-Return-to-Zero-Mark (en: non-return-to-zero-mark)
NW	negatives Fenster (en: negative window)
PAC	Paketiersteuerung (en: packet assembly controller)
PLOP	Betriebsverfahren in der Bitübertragungsschicht (en: physical layer operation procedure)
PW	positives Fenster (en: positive window)
RF	Funkfrequenz, Hochfrequenz (en: radio frequency)
SEC	Einfachfehlererkennung (en: single error correction)
SS	Suspend_State (en: Suspend_State)
TC	Fernsteuerung (en: telecommand)
TED	Dreifachfehlererkennung (en: triple error detection)
TT	Timeout_Type (en: Timeout_Type)

Die AC-, CB- und BD-Rahmen sind in Tabelle 6-3 definiert.

3.4 Konventionen

3.4.1 Bit 0, Bit 1, Bit $N-1$

Um jedes Bit in einem N -Bit-Feld zu identifizieren, ist das erste zu übertragende Bit im Feld (d. h. das in einer graphischen Darstellung am weitesten links gelegene) als Bit 0 definiert; das folgende Bit ist als Bit 1 definiert usw. bis zum Bit $N-1$.

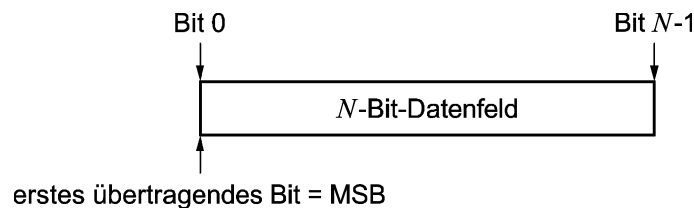


Bild 3-1 — Benummerungskonvention für Bits

3.4.2 Höchstwertiges Bit

Wird ein N -Bit-Feld zur Angabe eines Binärwerts genutzt (z. B. bei einer Zählvorrichtung), handelt es sich beim höchstwertigen Bit um das erste Bit im Feld, d. h. Bit 0 (siehe Bild 3-1).

3.4.3 Verwendung von Großbuchstaben bei den Bezeichnungen der Datenstrukturen und -felder

In dieser Norm werden für die Bezeichnungen der Datenstrukturen und -felder Großbuchstaben als erste Buchstaben verwendet.

Auf diese Weise können die Feldbezeichnungen innerhalb des Gesamttextes leicht identifiziert werden.

So ist beispielsweise das Feld „Frame Length“ („Rahmenlänge“) in einem Text, der Wörter wie „frame“ („Rahmen“) und „length“ („Länge“) enthält, leichter zu erkennen als das Feld „frame length“. Außerdem wird die Zweideutigkeit dahingehend vermeiden, wo die Bezeichnung beginnt und endet.

— Leerseite —

English version

Space engineering - Space data links - Telecommand protocols, synchronization and channel coding

Ingénierie spatiale - Liaisons de données spatiales -
Protocoles, synchronisation et codage canal des liaisons
télécommande

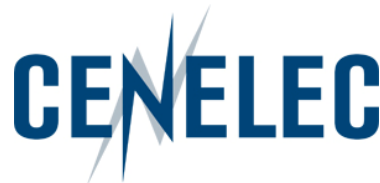
Raumfahrttechnik - Telekommando-Protokolle, -
synchronisation und -kanalkodierung

This European Standard was approved by CEN on 25 April 2014.

CEN and CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CEN and CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN and CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CEN and CENELEC members are the national standards bodies and national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.



CEN-CENELEC Management Centre:
Avenue Marnix 17, B-1000 Brussels

Table of contents

Foreword	10
1 Scope.....	11
2 Normative references	12
3 Terms, definitions and abbreviated terms.....	13
3.1 Terms from other standards.....	13
3.2 Terms specific to the present standard	13
3.3 Abbreviated terms.....	14
3.4 Conventions.....	15
3.4.1 Bit 0, bit 1, bit N-1.....	15
3.4.2 Most significant bit.....	15
3.4.3 Use of capitals for the names of data structures and fields.....	15
4 Overview.....	16
4.1 Presentation	16
4.2 Protocol profiles.....	16
4.3 Segmentation sublayer	17
4.4 Transfer sublayer.....	18
4.5 Synchronization and channel coding sublayer	18
4.6 Physical layer	18
4.7 Virtual channels	19
5 Segmentation sublayer	20
5.1 Overview	20
5.2 TC Segment	21
5.2.1 General	21
5.2.2 Segment Header	22
5.2.3 Segment Data Field	23
5.3 Transfer notification	23
5.3.1 Overview	23
5.3.2 Requirements.....	24
5.4 Blocking of packets.....	25

5.4.1	Overview	25
5.4.2	Virtual channels where TC Segments are used.....	25
5.4.3	Virtual channels where TC Segments are not used.....	26
5.4.4	Packet properties	26
5.4.5	Blocking function	27
5.4.6	Deblocking function.....	27
5.5	Segmentation	27
5.5.1	Overview	27
5.5.2	Segmenting function	28
5.5.3	Reassembly function.....	28
5.5.4	Packet assembly controller	29
5.6	MAP multiplexing	32
6	Transfer sublayer	33
6.1	Overview	33
6.1.1	Data structures in the transfer sublayer.....	33
6.1.2	Procedures in the transfer sublayer.....	33
6.2	TC Transfer Frame definition	35
6.2.1	General	35
6.2.2	Transfer Frame Primary Header.....	36
6.2.3	Transfer Frame Data Field	40
6.2.4	Frame Error Control Field.....	41
6.3	CLCW definition.....	43
6.3.1	General	43
6.3.2	Control Word Type	45
6.3.3	CLCW Version Number.....	45
6.3.4	Status Field	45
6.3.5	COP in Effect	45
6.3.6	Virtual Channel Identification.....	45
6.3.7	Reserved Spare	46
6.3.8	No RF Available Flag	46
6.3.9	No Bit Lock Flag.....	47
6.3.10	Lockout Flag	47
6.3.11	Wait Flag.....	48
6.3.12	Retransmit Flag.....	48
6.3.13	FARM-B Counter	48
6.3.14	Reserved Spare	48
6.3.15	Report Value	49

6.4	Frame header procedure	49
6.5	Frame error control procedure at the sending end	49
6.6	Frame delimiting and fill removal procedure.....	49
6.6.1	Overview	49
6.6.2	Actions	50
6.7	Frame error control procedure at the receiving end	50
6.8	Frame header validation procedure	50
6.8.1	Overview	50
6.8.2	Actions	51
6.9	Virtual channel multiplexing	51
6.9.1	Overview	51
6.9.2	Multiplexing mechanism	52
6.9.3	Demultiplexing	52
7	COP-1.....	53
7.1	Overview	53
7.1.1	Scope.....	53
7.1.2	Interfaces	53
7.1.3	Retransmission protocol.....	53
7.1.4	Frames.....	54
7.1.5	Services	54
7.1.6	Protocol machine	56
7.2	Internal variables	56
7.2.1	Overview.....	56
7.2.2	FOP-1 Variables	57
7.2.3	FARM-1 variables	64
7.3	Features of COP-1 interfaces	69
7.3.1	Overview.....	69
7.3.2	Parameters	70
7.4	Upper interface of COP-1 at the sending end	70
7.4.1	Overview.....	70
7.4.2	Sequence-controlled service management interface	71
7.4.3	Sequence-controlled service data transfer interface.....	77
7.4.4	Expedited service data transfer interface	80
7.5	Upper interface of COP-1 at the receiving end.....	82
7.5.1	Overview.....	82
7.5.2	Buffer management mechanism.....	82
7.5.3	The wait system	82

7.5.4	Single back-end buffer	83
7.5.5	FDU Arrived Indication	84
7.6	Lower interface of COP-1 at the sending end	84
7.6.1	Overview	84
7.6.2	Transmit Request for Frame signal	85
7.6.3	Abort request signal	85
7.6.4	Response signal	86
7.7	Lower interface of COP-1 at the receiving end.....	86
7.7.1	Overview	86
7.7.2	Valid Frame Arrived Indication	87
7.8	Format of COP-1 control commands	87
7.8.1	Overview	87
7.8.2	General	87
7.8.3	Unlock.....	87
7.8.4	Set V(R).....	88
7.9	Actions.....	88
7.9.1	Format of the state tables.....	88
7.9.2	FOP-1	89
7.9.3	FARM-1	96
8	Synchronization and channel coding sublayer	111
8.1	Overview	111
8.2	BCH codeblock.....	111
8.2.1	General	111
8.2.2	Information	112
8.2.3	Error Control	112
8.3	Communications link transmission unit (CLTU).....	113
8.3.1	General	113
8.3.2	Start Sequence	113
8.3.3	Encoded Data	114
8.3.4	Tail Sequence	114
8.4	Randomization procedure.....	115
8.4.1	Overview	115
8.4.2	General	115
8.4.3	Random sequence.....	115
8.4.4	Application of the randomizer	116
8.5	BCH codeblock encoding procedure.....	116
8.6	Fill bits	117

8.6.1	Overview	117
8.6.2	General	117
8.7	Channel logic at the receiving end	118
8.8	BCH codeblock decoding procedures	119
8.8.1	Overview	119
8.8.2	General	120
9	Physical layer.....	121
9.1	Overview	121
9.2	Physical layer data structures	121
9.2.1	Acquisition sequence	121
9.2.2	CLTU	122
9.2.3	Idle sequence.....	122
9.3	Physical layer procedures.....	122
9.3.1	Overview.....	122
9.3.2	Carrier modulation modes	122
9.3.3	Telecommand session	124
9.3.4	Physical layer operation procedure (PLOP)	124
Annex A	(informative) Frame error control	127
A.1	Overview	127
A.2	Encoding	127
A.3	Decoding	128
Annex B	(informative) Changes from ESA-PSS-04-107	129
B.1	Overview	129
B.2	Technical changes.....	129
Annex C	(informative) Differences from CCSDS recommendations.....	132
C.1	Overview	132
C.2	Differences	132
Annex D	(informative) Performance issues	135
D.1	Introduction.....	135
D.2	Performance components.....	136
D.3	Factors affecting frame rejection rate.....	136
D.3.1	Bit synchronization factor	136
D.3.2	CLTU Start Sequence factors	137
D.3.3	BCH Codeblock Factor	138
D.3.4	Tail Sequence factor	140

D.3.5	Frames and CLTUs.....	141
D.4	Factors affecting frame undetected error rate	143
Annex E (informative)	Mission configuration parameters	147
E.1	Introduction.....	147
E.2	Parameters of a physical channel	147
E.2.1	Overview	147
E.2.2	Fixed parameters	147
E.2.3	Length of the acquisition sequence	147
E.2.4	Physical layer operation procedure	148
E.2.5	Transfer Frame Version Number	148
E.2.6	Maximum length of a TC Transfer Frame	148
E.2.7	Virtual channels	148
E.2.8	Use of the expedited service	148
E.2.9	Multiplexing parameters	148
E.3	Parameters of a virtual channel	149
E.3.1	Overview	149
E.3.2	Spacecraft Identifier and Virtual Channel Identifier	149
E.3.3	Maximum length of a TC Transfer Frame	149
E.3.4	FOP-1 parameters	149
E.3.5	CLCW reporting rate	149
E.3.6	Status Field of CLCW	149
E.3.7	Fixed parameters	149
E.3.8	FARM-1 sliding window parameters	150
E.3.9	Use of TC Segments	150
E.3.10	Parameters of a virtual channel with TC Segments	150
E.3.11	Parameters of a virtual channel without TC Segments	150
E.4	Parameters of a MAP	151
E.4.1	Overview	151
E.4.2	MAP Identifier	151
E.4.3	Use of the blocking function	151
E.4.4	Segmentation function	151
E.5	Parameters for packet types	152
E.5.1	Overview	152
E.5.2	Valid packet version numbers	152
Bibliography.....		153

Figures

Figure 3-1: numbering convention	15
Figure 4-1: Layers and sublayers specified in this Standard	17
Figure 5-1: TC Segment	21
Figure 5-2: Example of blocking of packets	25
Figure 5-3: Example of segmentation of a user data unit	28
Figure 6-1: TC Transfer Frame format	36
Figure 6-2: Format of a CLCW	44
Figure 7-1: COP-1 sequence variables	55
Figure 7-2: FARM sliding window concept	67
Figure 7-3: State table format	89
Figure 7-4: Actions for look for directive	94
Figure 7-5: Actions for look for FDU	95
Figure 7-6: FOP-1 state transitions for main protocol	105
Figure 7-7: FOP-1 state transitions for initialisation protocol	106
Figure 7-8: FOP-1 state transitions	107
Figure 7-9: FARM-1 state transitions	110
Figure 8-1: BCH codeblock format	112
Figure 8-2: Format of a CLTU	113
Figure 8-3: Bit pattern of the Start Sequence	113
Figure 8-4: Bit transition generator logic diagram	116
Figure 8-5: (63,56) Modified BCH code generator	117
Figure 8-6: State diagram for the channel (receiving end)	119
Figure 9-1: Sequence of CMMs comprising PLOP-2	125
Figure 9-2: Sequence of CMMs comprising PLOP-1	126
Figure A-1 : Encoder	127
Figure A-2 : Decoder	128
Figure D-1 : Frame rejection probability, P_{FY} , in SEC mode using PLOP-2	143
Figure D-2 : Probability of undetected error in a frame in SEC mode	146

Tables

Table 6-1: Sending-end procedures in the transfer sublayer	34
Table 6-2: Receiving-end procedures in the transfer sublayer	35
Table 6-3: Combined Bypass Flag and Control Command Flag	38
Table 6-4: Fields in a CLCW	43
Table 7-1: COP-1 interfaces	70

Table 7-2: Signals for management interface	71
Table 7-3: FOP-1 directive types and qualifiers	73
Table 7-4: Reasons for an Alert notification	75
Table 7-5: Signals for sequence-controlled service data transfer interface	78
Table 7-6: Signals for expedited service data transfer interface.....	80
Table 7-7: Signals for the interface of FOP-1 to the lower procedures.....	84
Table 7-8: FOP-1 state table	98
Table 7-9: FARM-1 state table.....	108
Table 8-1: Channel states (receiving end)	118
Table 8-2: Channel events (receiving end)	119
Table 9-1: Carrier modulation modes	123
Table B-1 : Field name differences from ESA-PSS-04-107	131
Table B-2 : Names with “Communications” or “Command”	131
Table C-1 : Name differences from CCSDS recommendations	134
Table D-1 : Probability of not recognizing the Start Sequence	138
Table D-2 : Meaning of decoding values	138
Table D-3 : Decoding cases in SEC mode.....	138
Table D-4 : Probability of codeblock rejection for a CLTU during decoding in SEC mode...	140
Table D-5 : Parity and Syndrome when Tail Sequence has errors.....	141
Table D-6 : Probability of missing the Tail Sequence.....	141
Table D-7 : Frame rejection probability, P_{FY} (PLOP-2).....	142
Table D-8 : Sources of undetected errors (SEC mode).....	143
Table D-9 : Probability of n errors occurring in a codeblock	144
Table D-10 : Error detection performance when decoding a codeblock in SEC mode	144
Table D-11 : Probability of undetected error in a frame, SEC mode, with CRC	145

Foreword

This document (EN 16603-50-04:2014) has been prepared by Technical Committee CEN/CLC/TC 5 "Space", the secretariat of which is held by DIN.

This standard (EN 16603-50-04:2014) originates from ECSS-E-ST-50-04C.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by March 2015, and conflicting national standards shall be withdrawn at the latest by March 2015.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN [and/or CENELEC] shall not be held responsible for identifying any or all such patent rights.

This document has been prepared under a mandate given to CEN by the European Commission and the European Free Trade Association.

This document has been developed to cover specifically space systems and has therefore precedence over any EN covering the same scope but with a wider domain of applicability (e.g. : aerospace).

According to the CEN-CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

1

Scope

This Standard specifies the data structures and protocols for a telecommand space data link and the procedure for physical layer operation.

Usually, the source of data on a telecommand space data link is located on the ground and the receiver is located in space. However, the Standard may also be used for space-to-space telecommand data links.

Further provisions and guidance on the application of this standard can be found, respectively, in the following documents:

- The higher level standard ECSS-E-ST-50 “Communications”, which defines the principle characteristics of communication protocols and related services for all communication layers relevant for space communication (physical- to application-layer), and their basic relationship to each other.
- The handbook ECSS-E-HB-50 “Communications guidelines”, which provides information about specific implementation characteristics of these protocols in order to support the choice of a certain communications profile for the specific requirements of a space mission.

Users of this present standard are invited to consult these documents before taking decisions on the implementation of the present one.

This standard may be tailored for the specific characteristics and constraints of a space project in conformance with ECSS-S-ST-00.

2

Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this ECSS Standard. For dated references, subsequent amendments to, or revisions of any of these publications, do not apply. However, parties to agreements based on this ECSS Standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references the latest edition of the publication referred to applies.

EN reference	Reference in text	Title
EN 16601-00-01	ECSS-S-ST-00-01	ECSS system - Glossary of terms
	CCSDS 135.0-B-3 Issue 3, October 2006	Space Link Identifiers – Blue Book

3

Terms, definitions and abbreviated terms

3.1 Terms from other standards

For the purpose of this Standard, the terms and definitions from ECSS-S-ST-00-01 apply.

3.2 Terms specific to the present standard

3.2.1 mission phase

period of a mission during which specified telecommand characteristics are fixed

NOTE The transition between two consecutive mission phases can cause an interruption of the telecommand data link services.

3.2.2 octet

group of eight bits

NOTE 1 The numbering for octets within a data structure starts with 0.

NOTE 2 Refer to 3.4 for the convention for the numbering of bits.

3.2.3 packet

variable-length data structure consisting of higher layer user data encapsulated within standard header information

3.2.4 static

unchanged within a specific virtual channel

NOTE 1 This Standard contains requirements on the invariability, throughout one or all mission phases, of certain characteristics of the data structures specified in it.

NOTE 2 For virtual channel refer to 4.7.

3.2.5 physical layer operation procedure

procedure in the physical layer to activate and deactivate the physical communications channel by invoking RF carrier and modulation techniques

3.3 Abbreviated terms

For the purpose of this Standard, the abbreviated terms from ECSS-ST-00-01 and the following apply:

Abbreviation	Meaning
AD	acceptance check and data
ARQ	automatic repeat request
BC	bypass of acceptance check and control
BCH	Bose-Chaudhuri-Hocquenghem
BD	bypass of acceptance check and data
BER	bit error rate
BTG	bit transition generator
CCSDS	Consultative Committee for Space Data Systems
CLCW	communications link control word
CLTU	communications link transmission unit
CMM	carrier modulation mode
COP	communications operation procedure
FARM	frame acceptance and reporting mechanism
FDU	telecommand frame data unit
FECF	Frame Error Control Field
FOP	frame operation procedure
ID	identifier, or, identification
LLIF	lower layer interface
MAP	multiplexer access point
MSB	most significant bit
NRZ-M	non-return-to-zero-mark
NW	negative window
PAC	packet assembly controller
PLOP	physical layer operation procedure
PW	positive window
RF	radio frequency
SEC	single error correction
SS	Suspend_State
TC	telecommand
TED	triple error detection
TT	Timeout_Type

NOTE AC, CB and BD frames are defined in Table 6-3.

3.4 Conventions

3.4.1 Bit 0, bit 1, bit N-1

To identify each bit in an N-bit field, the first bit in the field to be transferred (i.e. the most left justified in a graphical representation) is defined as bit 0; the following bit is defined as bit 1 and so on up to bit N-1.

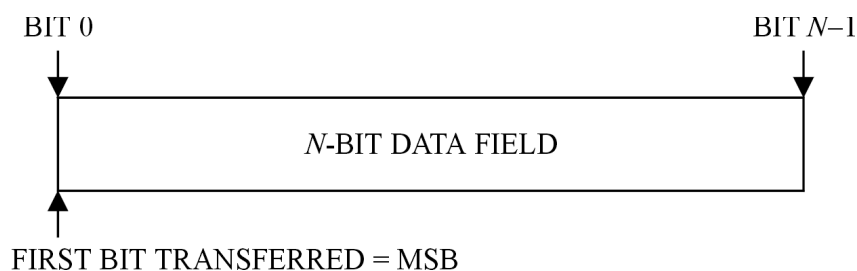


Figure 3-1: numbering convention

3.4.2 Most significant bit

When an N-bit field is used to express a binary value (such as a counter), the most significant bit is the first bit of the field, i.e. bit 0 (see Figure 3-1).

3.4.3 Use of capitals for the names of data structures and fields

In this Standard initial capitals are used for the names of data structures and fields.

This enables field names to be easily identified in the surrounding text.

For example, the field Frame Length is easier to see than frame length in text containing words such as frame and length. It also prevents ambiguity over where the name begins and ends.

4

Overview

4.1 Presentation

This Standard describes the data structures and protocols used in space missions to transmit data on telecommand space data links. It is designed to meet the requirements of space missions for the reliable transfer of telecommands over ground-to-space or space-to-space communications links.

Annex E describes the mission configuration parameters within the scope of this Standard.

The functions and protocols of the telecommand space link are grouped into layers and sublayers. The functions and protocols addressed in this Standard are grouped as follow:

- the segmentation sublayer,
- the transfer sublayer,
- the synchronization and channel coding sublayer, and
- the procedures in the physical layer at the sending end.

The segmentation sublayer, the transfer sublayer and the synchronization and channel coding sublayer are sublayers of the data link layer.

The layers and sublayers defined in this Standard provide a means for transferring data units on behalf of higher layers.

In general, the procedures, protocols and functions defined in this Standard are not concerned with the format, contents or applications of the higher layer data units. The only exception is the blocking function of the segmentation sublayer: the higher layer data units which use this function are restricted to certain packet types.

4.2 Protocol profiles

Figure 4-1 illustrates the following:

- The relationship of the layers and sublayers specified in this Standard.
- The data structures passed between the peer entities. The data structures are introduced in the overview clauses 4.3 to 4.5.

The protocol profile for a project can differ from the profile represented by Figure 4-1.

When defining the layers and sublayers in this Standard, reference is sometimes made to one of the other layers or sublayers, but such references are not intended to exclude differences from the profile represented by Figure 4-1.

The protocol profile in Figure 4-1 does not show any security sublayers. As part of the tailoring process described in Clause 1 a project with security requirements can insert security sublayers into the protocol profile of the telecommand space data link.

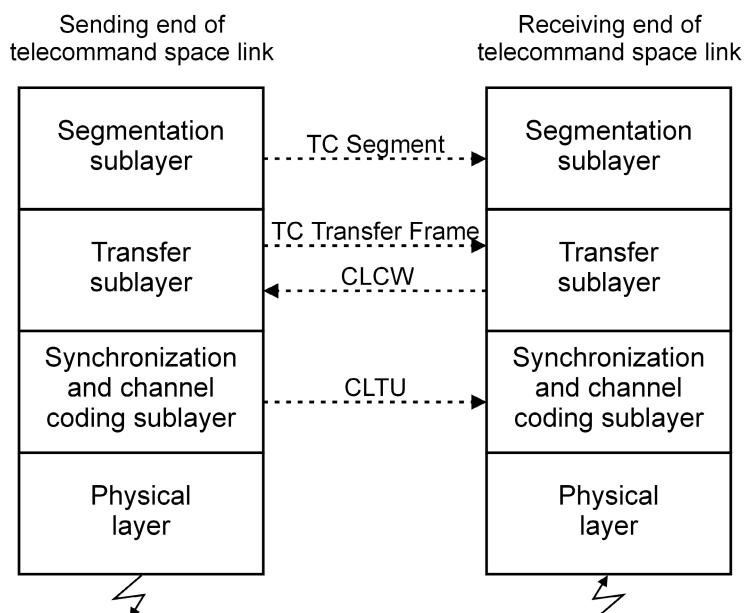


Figure 4-1: Layers and sublayers specified in this Standard

4.3 Segmentation sublayer

The TC segment contains fields to support two of the functions of the segmentation sublayer:

- the segmentation function, and
- a multiplexing function.

The segmentation function breaks long data units into smaller portions so that they are consistent with the limited length of the frames in the transfer sublayer. At the receiving end, the segmentation function reassembles the portions into the original data units.

The segmentation sublayer also includes a blocking function for packets so that multiple packets can be carried in a frame.

Clause 5 provides the specification of the segmentation sublayer and of the TC segment.

4.4 Transfer sublayer

The TC Transfer Frame is a variable-length data structure, designed for the efficient transfer of variable-length data units.

The transfer sublayer contains procedures for generating and processing TC Transfer Frames. The procedures include a multiplexing function.

The procedures are defined in Clause 6, which also includes the definition of the TC Transfer Frame.

The transfer sublayer includes the communications operation procedure (COP-1), which operates the transmission protocol of the transfer sublayer. The detailed specification of COP-1 is given in Clause 7.

COP-1 supports two service types:

- the sequence-controlled service, and
- the expedited service.

COP-1 uses protocol status information which is passed from the transfer sublayer at the receiving end to the transfer sublayer at the sending end. The information is contained in a communications link control word (CLCW), which can be transmitted in a telemetry transfer frame.

4.5 Synchronization and channel coding sublayer

The synchronization and channel coding sublayer establishes an error-controlled data channel through which data can be transferred. The data is encoded, using a block code with error-correcting capabilities, to reduce the effects of noise on the data in the space channel.

The synchronization and channel coding sublayer includes procedures for information bit randomizing, to ensure frequent bit transitions in the communications channel. Synchronization for the codeblock and delimiting of the beginning of user data are provided by the communications link transmission unit (CLTU) data structure.

Clause 8 defines the synchronization and channel coding sublayer and also contains the definition of the CLTU.

In Annex D the performance of the coding and data structures in the synchronization and channel coding sublayer is discussed and the related frame error rates in the transfer sublayer.

4.6 Physical layer

The physical layer is the lowest layer of the telecommand space link. In this Standard the physical layer is designed to work together with the synchronization and channel coding sublayer. It accepts CLTUs from the synchronization and channel coding sublayer at the sending end. At the receiving end it supplies the synchronization and channel coding sublayer with a symbol stream containing the CLTUs.

Clause 9 defines the physical layer procedures, including the associated data structures.

Other aspects of the physical layer, such as radio frequency characteristics, are outside the scope of this Standard.

4.7 Virtual channels

The TC Transfer Frame supports the division of the physical channel into multiple logically-separate virtual channels by means of identifier fields in the header of a TC Transfer Frame. The identifier fields are the Virtual Channel Identifier and the Spacecraft Identifier. The virtual channel multiplexing is a procedure of the transfer sublayer.

This Standard does not specify the manner in which the virtual channels are multiplexed into a physical channel: the Spacecraft Identifier can be included as part of the multiplexing process in a given implementation. Clause 6.9 defines virtual channel multiplexing.

Segmentation sublayer

5.1 Overview

The segmentation sublayer accepts variable-length octet-aligned data units from the next higher layer or sublayer at the sending end and delivers them to the next higher layer or sublayer at the receiving end. Within the segmentation sublayer these data units are referred to as user data units.

The segmentation sublayer provides three functions: blocking, segmentation and multiplexing.

- The blocking function combines multiple user data units so that they are carried in a single frame of the transfer sublayer. The user data units for the blocking function are limited to packets.
- The segmentation function breaks long user data units into multiple shorter portions at the sending end and reassembles them at the receiving end.
- User data units allocated to different multiplexer access points (MAPs) are multiplexed together so that they share one virtual channel of the transfer sublayer.

The principal data structure in the segmentation sublayer is the TC Segment. This is a variable-length data structure containing an integral number of octets. The TC Segment has fields in its header to support segmentation and MAP multiplexing.

If segmentation or MAP multiplexing is used for a virtual channel, then the virtual channel uses TC Segments.

If neither segmentation nor MAP multiplexing is used for a virtual channel, then the use of TC Segments on the virtual channel is optional. A physical channel can have some virtual channels that use TC Segments and some virtual channels that do not.

At the sending end, the data units generated by the segmentation sublayer are passed to the next lower sublayer:

- It is assumed that the next lower sublayer accepts variable-length data units from the segmentation sublayer.
- The next lower sublayer can set maximum length limitations for the data units it accepts from the segmentation sublayer.
- When the segmentation function is in use, it is also assumed that the next lower sublayer can provide a service which ensures that the data units are delivered to the receiving end in sequence and without gaps or duplication.

5.2 TC Segment

5.2.1 General

- a. The length of the TC Segment shall not exceed the maximum length accepted by the next lower sublayer below the segmentation sublayer.

NOTE 1 The TC Segment is a variable-length data structure.

NOTE 2 The maximum length accepted by the transfer sublayer can vary across different virtual channels.

- b. The length of a TC Segment shall be an integer number of octets.
- c. A TC Segment shall consist of two fields, positioned contiguously, in the following sequence:

1. Segment Header 8 bits
2. Segment Data Field variable length

NOTE The format of the TC Segment is shown in Figure 5-1.

- d. If the next lower sublayer below the segmentation sublayer is the transfer sublayer, and the segmentation sublayer at the sending end is generating TC Segments for a particular virtual channel, then all data units delivered by the transfer sublayer at the receiving end for that virtual channel shall be TC Segments.

NOTE TC Segments do not share a virtual channel with other data structures.

- e. If the next lower sublayer below the segmentation sublayer is the transfer sublayer, then each TC Segment shall be placed in the Transfer Frame Data Field of a TC Transfer Frame such that the Transfer Frame Data Field contains exactly one complete TC Segment.

NOTE In this case, the length of the Transfer Frame Data Field is equal to the length of the TC Segment.

SEGMENT HEADER		SEGMENT DATA FIELD
SEQUENCE FLAGS	MAP IDENTIFIER	
2 bits	6 bits	variable

Figure 5-1: TC Segment

5.2.2 Segment Header

5.2.2.1 General

- a. The Segment Header shall always be present in a TC Segment.
- b. The Segment Header shall be contained in bits 0-7 of the TC Segment.
- c. The Segment Header shall consist of two fields, positioned contiguously, in the following sequence:
 1. Sequence Flags 2 bits
 2. MAP Identifier 6 bits

NOTE Both fields are always present in a Segment Header.

5.2.2.2 Sequence Flags

- a. The Sequence Flags shall always be present in a Segment Header.
- b. The Sequence Flags shall be contained in bits 0-1 of the Segment Header.
- c. The Sequence Flags shall be set as follows:
 1. '01' for the first segment of a user data unit;
 2. '00' for a continuing segment of a user data unit;
 3. '10' for the last segment of a user data unit;
 4. '11' no segmentation.

NOTE The Sequence Flags provide information about the sequential position of the segment to enable the receiving end to reassemble the user data units from a stream of segments. However, the flags do not provide a sequence count, so the reassembly process depends on the stream of segments being delivered in sequence and without omissions. The sequence-controlled service of the transfer sublayer can provide delivery in sequence and without omissions.

5.2.2.3 MAP Identifier

- a. The MAP Identifier shall always be present in a Segment Header.
- b. The MAP Identifier shall be contained in bits 2-7 of the Segment Header.

NOTE The six-bit MAP Identifier enables up to 64 MAPs, from MAP 0 to MAP 63, to be associated with each virtual channel of the transfer sublayer.

- c. The MAP Identifier shall provide the identification of the MAP to which the TC Segment belongs.

NOTE 1 There are no restrictions on the selection of MAP Identifiers. The MAPs need not be

numbered consecutively. However, if the packet assembly controller defined in clause 5.5.4 is in use, then MAPs are handled in pairs and the additional requirements specified in clause 5.5.4.2 apply.

NOTE 2 When a user data unit is placed in a sequence of TC Segments, then all the TC Segments for the user data unit have the same value for the MAP Identifier.

- d. If MAP multiplexing is not used on a particular virtual channel of the transfer sublayer, the MAP Identifier shall have a constant value for all TC Segments on that virtual channel.

NOTE In this case the virtual channel has a single MAP.

5.2.3 Segment Data Field

- a. The Segment Data Field shall always be present in a TC Segment, except as specified in 5.2.3e.
- b. The Segment Data Field shall start at bit 8 of the TC Segment.
- c. The maximum length of the Segment Data Field is one octet less than the maximum length of a TC Segment.

NOTE The length of the Segment Data Field is variable.

- d. The length of the Segment Data Field shall be an integer number of octets.
- e. If the packet assembly controller defined in clause 5.5.4 is in use, then the Segment Data Field may be absent from some TC Segments.

NOTE When the packet assembly controller is in use, some MAPs can carry control segments. A control segment is a special case of a TC Segment that has no Segment Data Field.

5.3 Transfer notification

5.3.1 Overview

The sequence-controlled service of the COP-1 protocol of the transfer sublayer uses acknowledgements to obtain positive confirmation that data units have arrived at the transfer layer at the receiving end. The data units are the frame data units carried in type-A TC Transfer Frames.

In the transfer notification (AD service) signal, COP-1 uses the Positive Confirm and Negative Confirm responses to provide information about the delivery status of a data unit. The transfer notification signal is defined in clause 7.4.3.3, along with explanations of the meaning of the confirm responses.

The segmentation sublayer at the sending end takes the information obtained from the confirm responses and makes it available to the next higher layer or sublayer. If a user data unit is carried in multiple segments, the segmentation sublayer combines the confirm responses for the individual segments.

The mechanism by which the segmentation sublayer makes the information available to the next higher layer or sublayer is outside the scope of this Standard.

The handling by the segmentation sublayer at the sending end of other responses from the transfer notification (AD service) signal, or of information provided by other COP-1 signals, is outside the scope of this Standard.

5.3.2 Requirements

- a. If a user data unit of the segmentation sublayer is transferred using the sequence-controlled service of the transfer sublayer, then the segmentation sublayer at the sending end shall provide, to the next higher layer or sublayer, the information about the delivery of the user data unit specified in clauses 5.3.2b to 5.3.2d.
- b. If a user data unit is not divided into multiple segments, then the segmentation sublayer shall provide a Positive Confirm or Negative Confirm to the next higher layer or sublayer, according to the response contained in the COP-1 signal relating to the transfer of the user data unit.
- c. If a user data unit is divided into multiple segments, then the segmentation sublayer shall provide a Positive Confirm to the next higher layer or sublayer if COP-1 has signalled Positive Confirm responses for all the segments.
- d. If a user data unit is divided into multiple segments, then the segmentation sublayer shall provide a Negative Confirm to the next higher layer or sublayer if COP-1 has signalled a Negative Confirm response for one or more of the segments.

NOTE If some segments of a user data unit have been signalled by a Positive Confirm but at least one segment has been signalled by a Negative Confirm, then the segmentation sublayer provides a Negative Confirm for the user data unit.

5.4 Blocking of packets

5.4.1 Overview

Blocking of packets is performed by a blocking function at the sending end and a deblocking function at the receiving end.

The blocking function blocks multiple packets into a single data unit. The function is constrained by the limitations on the length of the data unit. Figure 5-2 illustrates an example.

Additional constraints can apply to the way the segmentation sublayer uses the blocking function.

For example, an implementation can set a limit for the delay imposed on a packet while waiting for more packets to be blocked with it. The additional constraints and the mechanisms for ensuring transmission of all packets are implementation dependent and outside the scope of this Standard.

The deblocking function uses fields in the standard packet headers in order to extract the packets. The function therefore depends on knowledge of the position, size and meaning of certain fields in the headers.

Blocking can be used with or without TC Segments.

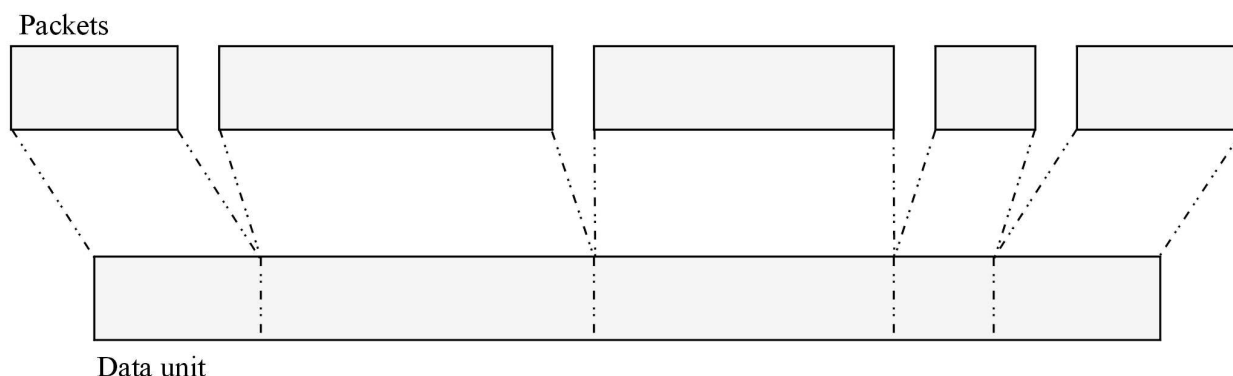


Figure 5-2: Example of blocking of packets

5.4.2 Virtual channels where TC Segments are used

- a. The blocking and deblocking functions shall be conducted independently for each MAP.
- b. The length of the data unit resulting from blocking shall not exceed the maximum length of the Segment Data Field of a TC Segment.
- c. A virtual channel may have some MAPs where the blocking of packets is enabled and some MAPs where the blocking of packets is not enabled.
- d. If the blocking of packets is enabled for a MAP, then all data units for that MAP, that are passed to the segmentation sublayer from the next higher layer or sublayer at the sending end, shall be packets with the properties defined in clause 5.4.4.

- e. If the blocking function blocks multiple packets into a data unit, then the Sequence Flags of the TC Segment which carries the data unit shall be set to indicate no segmentation.

NOTE A data unit containing multiple packets is not divided into multiple segments by the segmentation function.

5.4.3 Virtual channels where TC Segments are not used

- a. The blocking and deblocking functions shall be conducted independently for each virtual channel.
- b. The length of the data unit resulting from blocking shall not exceed the maximum length of the data units accepted by the next lower sublayer below the segmentation sublayer.
- c. A physical channel may have some virtual channels where the blocking of packets is enabled and some virtual channels where the blocking of packets is not enabled.
- d. If the blocking of packets is enabled for a virtual channel, then all data units for that virtual channel, which are passed to the segmentation sublayer from the next higher layer or sublayer at the sending end, shall be packets with the properties defined in clause 5.4.4.

5.4.4 Packet properties

- a. A packet handled by the blocking function shall have a defined Packet Version Number in conformance with CCSDS 135.0-B-3 clause 7.6 and with the definition of the related data structure specified in the same clause.

NOTE 1 The Packet Version Number occupies the first three bits of the packet header.

NOTE 2 CCSDS 135.0-B-3 clause 7.6 contains a list of the defined Packet Version Numbers, which are also referred to as authorized Packet Version Numbers. It provides also references to the documents that specify the related packet data structures.

NOTE 3 For missions that use the blocking function, it is the responsibility of the mission designer to verify the availability of support by the telecommand transfer services for each defined Packet Version Number to be used by the mission.

5.4.5 Blocking function

- a. Multiple complete packets may be placed contiguously into a data unit.

NOTE 1 The blocking function is only applied to complete packets.

NOTE 2 Due to other constraints, such as implementation-dependent time limits, the segmentation sublayer can choose not to block some packets.

- b. The order of the packets shall not be changed to improve the blocking of the packets into the data units.

NOTE The order of arrival of the packets is preserved.

- c. Packets with different Packet Version Numbers may be blocked into the same data unit.

- d. If the blocking function processes packets for the sequence-controlled service and packets for the expedited service, then all packets that are blocked into the same data unit shall be for the same service.

NOTE The transfer sublayer supports the sequence-controlled service and the expedited service.

5.4.6 Deblocking function

- a. The deblocking function shall extract the packets from the data units it receives.

- b. To perform the extraction, the deblocking function shall use the packet length fields, together with the length of the received data unit.

NOTE The deblocking function inspects the Packet Version Number of each packet in order to locate and interpret the packet length field.

5.5 Segmentation

5.5.1 Overview

A segmenting function at the sending end breaks long user data units into multiple portions which are placed in a sequence of TC Segments. At the receiving end, a reassembly function reassembles the user data units. The functions are conducted independently for each MAP.

All TC Segments belonging to a specific user data unit have the same value for the MAP Identifier and use the same service (sequence-controlled service or expedited service) in the transfer sublayer.

Successful operation of the segmentation function relies on the stream of TC Segments being delivered in sequence and without omissions. The sequence-controlled service can provide delivery in sequence and without omissions.

Figure 5-3 illustrates an example of segmentation of a user data unit.

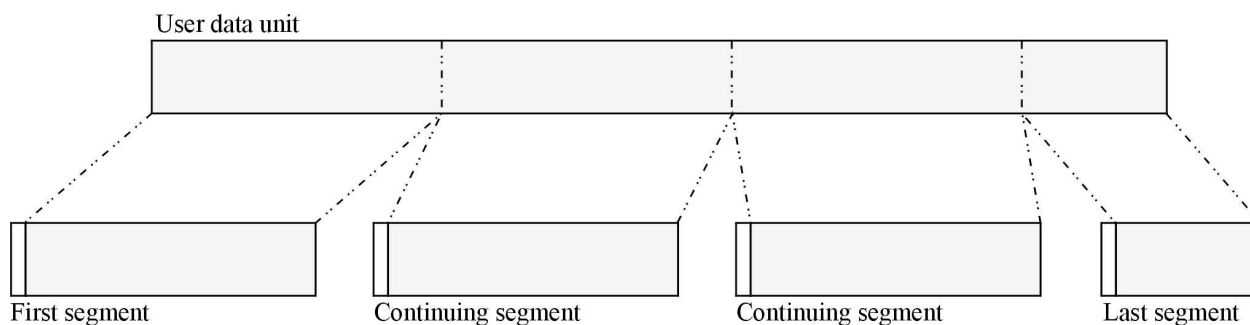


Figure 5-3: Example of segmentation of a user data unit

5.5.2 Segmenting function

- a. The segmenting function shall be conducted independently for each MAP.
- b. If a user data unit exceeds the maximum length of a Segment Data Field, the segmenting function shall split it as follows:
 1. For each portion, select the length of the portion such that it is an integer number of octets that does not exceed the maximum length of a Segment Data Field.
 2. Place the portions in order into a sequence of TC Segments.
 3. Place the first octet of the user data unit in the first octet of the Segment Data Field of the first TC Segment.
 4. Set the Sequence Flags of each TC Segment in the sequence to indicate whether it is a first, continuing or last segment.

NOTE 1 The Segment Data Field in the first and continuing TC Segments can have a length equal to the maximum length of the Segment Data Field. The Segment Data Field in the last TC Segment has a length equal to the residue of the user data unit.

NOTE 2 If a user data unit does not exceed the maximum length of a Segment Data Field, the complete user data unit is placed in a single TC Segment. In this case the Sequence Flags are set to indicate no segmentation.

5.5.3 Reassembly function

- a. The reassembly function shall be conducted independently for each MAP.
- b. The reassembly function shall reassemble the Segment Data Fields from a sequence of TC Segments to recreate the original user data unit.
- c. To perform the reassembly, the reassembly function shall use the Sequence Flags together with the lengths of the TC Segments.

NOTE 1 If a failure occurs during reassembly, the actions are undefined. The choice of whether

to deliver the incomplete or faulty user data unit to the next higher layer or sublayer is implementation dependent.

NOTE 2 The optional packet assembly controller defined in clause 5.5.4 specifies additional actions for the reassembly function, including the behaviour in the event of a reassembly failure.

5.5.4 Packet assembly controller

5.5.4.1 Overview

The packet assembly controller (PAC) can form part of the reassembly function at the receiving end of the segmentation function.

The reassembly function reassembles the segments for each MAP connection, in order to recreate the original user data units. The packet assembly controller carries out the reassembly, including the handling of exceptions.

When the PAC detects an exception it enters a lockout state. In the lockout state, the PAC does not reassemble user data units nor pass user data units to the next higher layer or sublayer. When it receives a MAP reset command, the PAC exits lockout state.

Despite the word “packet” in its name, the actions of the PAC apply to all user data units carried in TC Segments. The actions are not limited to any specific data structures. The name “packet assembly controller” is inherited from earlier standards.

If the segmentation function is not in operation, then the PAC is not used.

5.5.4.2 PAC for a pair of MAPs

- a. There shall be one PAC for each pair of MAP connections.
- b. A pair of MAP connections shall consist of one MAP for data and one MAP for control.
- c. The MAP Identifier for the data MAP shall have the most significant bit set to ‘0’.
- d. The MAP Identifier for the control MAP shall have the same value as the data MAP except that the most significant bit shall be set to ‘1’.
- e. A TC Segment carrying user data shall have the MAP Identifier of the data MAP.
- f. A TC Segment which is a PAC control segment shall have the MAP Identifier of the control MAP.

NOTE 1 The MAP Identifier is a 6-bit value, carried in the Segment Header. Therefore, for a pair of MAPs:

- the data MAP has an identifier in the range 0 to 31;

- the control MAP has an identifier in the range 32 to 63;
- control MAP Identifier = data MAP Identifier + 32;
- the least significant 5 bits of the two MAP Identifiers are the same.

NOTE 2 The PAC requirements on the use of the MAP Identifier apply in addition to the MAP Identifier specification in clause 5.2.2.3.

5.5.4.3 Reassembly of user data units

- a. The PAC reassembly function shall use the Sequence Flags in the Segment Header of each TC Segment to reassemble the user data units.

NOTE 1 The reassembly function implies that the number of octets in each segment is known by the PAC.

NOTE 2 The PAC reassembly function does not use any knowledge of the structure of the user data units.

For example, it does not make use of any length fields contained in the user data units. If the user data units have length fields, then verification that the length of the reassembled data unit is consistent with the value in the length field is outside the scope of this Standard.

5.5.4.4 PAC state

- a. The PAC shall enter a lockout state when it detects one of the following errors:
1. an incorrect sequence of data segments, as indicated by the Sequence Flags;
 2. a control segment with an invalid format.
- b. When the PAC is in a lockout state, it shall not reassemble user data units nor pass user data units to the next higher layer or sublayer.
- c. When the PAC is in a lockout state, it shall remain in that state until it receives a valid control segment as specified in clause 5.5.4.6.

NOTE 1 The following is a list of the incorrect sequences of segments that causes the PAC to enter lockout state. The values for the Sequence Flags are shown in parentheses.

- A first segment ('01') followed by a first segment ('01').
- A first segment ('01') followed by a no segmentation ('11').

- A continuing segment ('00') followed by a first segment ('01').
- A continuing segment ('00') followed by a no segmentation ('11').
- A last segment ('10') followed by a continuing segment ('00').
- A last segment ('10') followed by a last segment ('10').
- A no segmentation ('11') followed by a continuing segment ('00').
- A no segmentation ('11') followed by a last segment ('10').

NOTE 2 The lockout state of the PAC is unrelated to the FARM-1 Lockout state defined in clause 7.2.3.2.

5.5.4.5 PAC status report

- a. The PAC shall have a reassembly status flag set as follows:
 1. '0' when the PAC has completed reassembly of a user data unit;
 2. '1' when reassembly of a user data unit is in progress in the PAC.
- b. The PAC shall have a lockout status flag set as follows:
 1. '1' when the PAC is in a lockout state;
 2. '0' when the PAC is not in a lockout state.
- c. The PAC shall report its status to the sending end, including the following:
 1. the MAP Identifier of the data MAP, and
 2. the reassembly status flag, and
 3. the lockout status flag.

NOTE The correct operation of the PAC relies on the status of the PAC being known by the sending end. This Standard does not specify the format of the status information nor the mechanism to be used to transport it from the PAC to the appropriate entity at the sending end. It also does not specify any resulting behaviour at the sending end, such as the decision to send a control segment.

5.5.4.6 Control segment

- a. A control segment shall have a length of one octet.

NOTE A control segment is a special case of a TC Segment. It has no Segment Data Field and therefore consists of a Segment Header only.

- b. The Sequence Flags in the Segment Header of a control segment shall be set to '11'.
- c. The MAP Identifier in the Segment Header of a control segment shall contain the MAP Identifier of a control MAP.
- d. A valid control segment shall be considered to be a MAP reset command.

NOTE 1 If the PAC receives a segment that has the MAP Identifier of a control MAP but the segment does not conform to the format rules the PAC enters lockout state.

NOTE 2 This Standard does not specify any type of service (sequence-controlled service or expedited service) to be used for the transfer of a control segment.

5.5.4.7 MAP reset actions

- a. When the PAC receives a MAP reset command and the PAC reassembly status flag is '1', the PAC shall:
 - 1. discard the partially reassembled data unit, and
 - 2. set the reassembly status flag to '0'.
- b. When the PAC receives a MAP reset command and the PAC is in lockout state, the PAC shall exit lockout state.

NOTE The MAP reset command is used, for example, to recover from breaks in the sequence of TC Segments due to link difficulties or unplanned termination of transfer services.

5.6 MAP multiplexing

User data units allocated to different MAPs are multiplexed together so that they can share a virtual channel of the transfer sublayer. Each virtual channel can have up to 64 distinct MAPs (from MAP 0 to MAP 63) associated with it.

The MAP multiplexer selects a TC Segment depending on the priorities of the MAPs and the algorithm in use. This Standard does not specify any multiplexing algorithms. The type of priority scheme employed and the allocation of priorities to the individual MAPs is implementation dependent.

Although the multiplexing algorithms are implementation dependent, cross-support missions can specify the use of algorithms described in CCSDS 912.3-B-1.

If MAP multiplexing is in use, the order of arrival of user data units is not preserved across MAPs.

For example, a user data unit on a high-priority MAP can be processed before an earlier data unit on a low-priority MAP. Also, segments from user data units on different MAPs can be interleaved.

6

Transfer sublayer

6.1 Overview

6.1.1 Data structures in the transfer sublayer

The principal data structure in the transfer sublayer is the TC Transfer Frame, specified in clause 6.2. At the sending end, the transfer sublayer delivers TC Transfer Frames to the next lower sublayer.

The transfer sublayer accepts variable-length data units from the next higher sublayer at the sending end and delivers them to the next higher sublayer at the receiving end. Within the transfer sublayer, these data units are referred to as frame data units (FDUs) because each one is carried in the data field of a TC Transfer Frame. The transfer sublayer imposes constraints on the lengths of the data units.

The transfer sublayer at the receiving end accepts variable-length blocks containing an integral number of octets from the next lower sublayer. The transfer sublayer processes each received block as a candidate frame, to verify that it contains a valid TC Transfer Frame.

The operation of the transfer sublayer protocol uses status information which is passed from the receiving end to the sending end of the sublayer. The information is contained in a communications link control word (CLCW), which can be transmitted in a telemetry transfer frame. The format of the CLCW is defined in clause 6.3, and the processing for CLCWs in Clause 7.

6.1.2 Procedures in the transfer sublayer

6.1.2.1 Communications operation procedure

The transfer sublayer includes the communications operation procedure (COP), which operates the transmission protocol of the sublayer. A COP consists of a pair of synchronized procedures which execute within one virtual channel at the sending and receiving ends of the sublayer. This Standard defines one COP, called COP-1, whose detailed specification is contained in Clause 7. COP-1 supports two service types for data transfer: the sequence-controlled service and the expedited service.

6.1.2.2 Sending end

Table 6-1 shows the procedures in the transfer sublayer at the sending end.

At the sending end, the first procedure in the transfer sublayer is the frame operation procedure (FOP-1), which is the sending end of COP-1.

The COP-1 procedures operate within a virtual channel so the transfer sublayer has an instance of FOP-1 for each virtual channel.

There are two procedures to complete the fields in each Transfer Frame:

- the frame header procedure places values in the fields of the Transfer Frame Primary Header, and
- the frame error control procedure then generates the frame check sequence and places it in the Frame Error Control Field.

The transfer sublayer includes virtual channel multiplexing. Because FOP-1 has an instance for each virtual channel, the virtual channel multiplexing is after FOP-1, but its position relative to the other two procedures is implementation dependent.

Table 6-1: Sending-end procedures in the transfer sublayer

Procedure	Clause	Position in the sequence of execution	(Peer procedure at the receiving end)
FOP-1	7	First procedure in the transfer sublayer	(FARM-1)
Frame header procedure	6.4	<ul style="list-style-type: none">- After FOP-1- Before frame error control procedure	(Frame header validation procedure)
Frame error control procedure	6.5	After frame header procedure	(Frame error control procedure)
Virtual channel multiplexing	6.9	After FOP-1	(Virtual channel demultiplexing)

6.1.2.3 Receiving end

Table 6-2 shows the procedures in the transfer sublayer at the receiving end.

At the receiving end, the first procedure is the frame delimiting and fill removal procedure. This validates the length and removes any fill data from the end of each candidate frame.

The frame error control procedure verifies the frame check sequence in the Frame Error Control Field. The frame header validation procedure checks the remaining fields of the Transfer Frame Primary Header.

The last procedure is the frame acceptance and reporting mechanism (FARM-1), which is the receiving end of COP-1. The COP-1 procedures operate within a

virtual channel so the transfer sublayer has an instance of FARM-1 for each virtual channel.

The transfer sublayer includes virtual channel demultiplexing, which is after the frame delimiting and fill removal procedure. Because FARM-1 has an instance for each virtual channel, the virtual channel demultiplexing is before FARM-1, but its position relative to the other two procedures is implementation dependent.

Table 6-2: Receiving-end procedures in the transfer sublayer

Procedure	Clause	Position in the sequence of execution	(Peer procedure at the sending end)
Frame delimiting and fill removal procedure	6.6	First procedure in the transfer sublayer	(none)
Frame error control procedure	6.7	- After frame delimiting and fill removal procedure - Before frame header validation procedure	(Frame error control procedure)
Frame header validation procedure	6.8	- After frame error control procedure - Before FARM-1	(Frame header procedure)
Virtual channel demultiplexing	6.9	- After frame delimiting and fill removal procedure - Before FARM-1	(Virtual channel multiplexing)
FARM-1	7	Last procedure in the transfer sublayer	(FOP-1)

6.2 TC Transfer Frame definition

6.2.1 General

- a. The TC Transfer Frame shall consist of the major fields, positioned contiguously, in the following sequence:

1. Transfer Frame Primary Header 5 octets
2. Transfer Frame Data Field variable length
3. Frame Error Control Field 2 octets

NOTE 1 All three fields are always present in a TC Transfer Frame.

NOTE 2 Figure 6-1 illustrates the detailed format of the TC Transfer Frame.

- b. The length of a TC Transfer Frame shall be an integer number of octets.

NOTE 1 The TC Transfer Frame is a variable-length data structure.

NOTE 2 The length of a TC Transfer Frame is indicated by the Frame Length field defined in clause 6.2.2.8. Because of constraints imposed by the Frame Length field, the length of a TC Transfer Frame does not exceed 1024 octets. For a mission, a value of less than 1024 octets can be selected for the length of the longest TC Transfer Frame that can be carried on a particular virtual channel.

NOTE 3 The length of the Transfer Frame Data Field defined in clause 6.2.3 is not less than 1 octet. As a result, a TC Transfer Frame has a length which is not less than 8 octets.

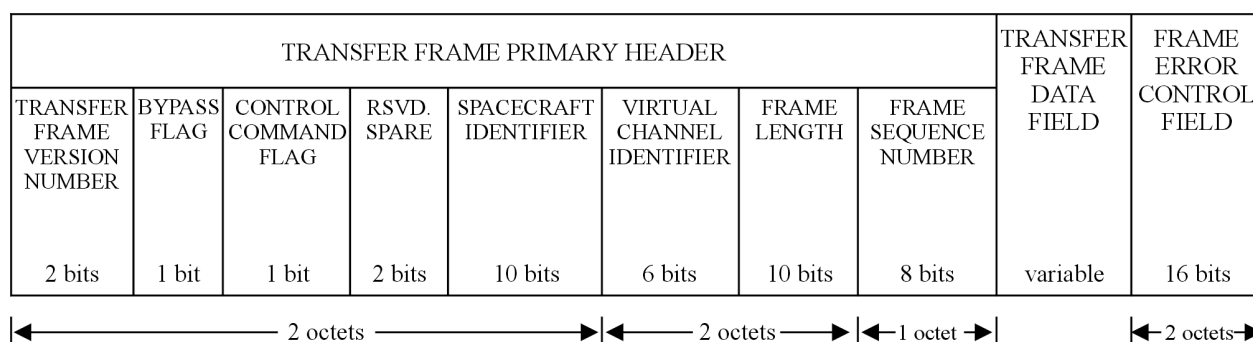


Figure 6-1: TC Transfer Frame format

6.2.2 Transfer Frame Primary Header

6.2.2.1 General

- a. The Transfer Frame Primary Header shall always be present in a TC Transfer Frame.
- b. The Transfer Frame Primary Header shall consist of eight fields, positioned contiguously, in the following sequence:
- Transfer Frame Version Number 2 bits
 - Bypass Flag 1 bit
 - Control Command Flag 1 bit
 - Reserved Spare 2 bits
 - Spacecraft Identifier 10 bits
 - Virtual Channel Identifier 6 bits

- | | | |
|----|-----------------------|---------|
| 7. | Frame Length | 10 bits |
| 8. | Frame Sequence Number | 8 bits |

NOTE All eight fields are always present in a Transfer Frame Primary Header.

6.2.2.2 Transfer Frame Version Number

- a. The Transfer Frame Version Number shall always be present in a Transfer Frame Primary Header.
- b. The Transfer Frame Version Number shall be contained in bits 0-1 of the Transfer Frame Primary Header.
- c. The Transfer Frame Version Number shall be set to the binary value '00'.

NOTE The value '00' defines version 1 of the TC Transfer Frame. Other values of this field are reserved.

6.2.2.3 Bypass Flag

- a. The Bypass Flag shall always be present in a Transfer Frame Primary Header.
- b. The Bypass Flag shall be contained in bit 2 of the Transfer Frame Primary Header.
- c. The Bypass Flag shall be set as follows:
 1. '0' for a type-A Transfer Frame;
 2. '1' for a type-B Transfer Frame.
- d. The Bypass Flag shall be interpreted in combination with the Control Command Flag as defined in Table 6-3.

NOTE 1 When the Bypass Flag is set to '1', the TC Transfer Frame bypasses the sequence control checks at the receiving end of the COP-1 procedures.

NOTE 2 To operate the sequence-controlled service, a virtual channel carries type-A and type-B Transfer Frames.

6.2.2.4 Control Command Flag

- a. The Control Command Flag shall always be present in a Transfer Frame Primary Header.
- b. The Control Command Flag shall be contained in bit 3 of the Transfer Frame Primary Header.
- c. The Control Command Flag shall be set as follows:
 1. '0' for a type-D Transfer Frame, carrying data;
 2. '1' for a type-C Transfer Frame, carrying a control command.

- d. The Control Command Flag shall be interpreted in combination with the Bypass Flag as defined in Table 6-3.

NOTE A control command is a protocol control instruction used in COP-1 to configure the procedures at the receiving end.

Table 6-3: Combined Bypass Flag and Control Command Flag

Bypass Flag	Control Command Flag	Interpretation
0	0	AD frame, carrying data in the sequence-controlled service
0	1	Reserved for future application
1	0	BD frame, carrying data in the expedited service
1	1	BC frame, carrying a control command

6.2.2.5 Reserved Spare

- a. The spare field contained in bits 4-5 of the Transfer Frame Primary Header shall be set to '00'.

NOTE The use of this spare field is reserved for future applications.

6.2.2.6 Spacecraft Identifier

- a. The Spacecraft Identifier shall always be present in a Transfer Frame Primary Header.
- b. The Spacecraft Identifier shall be contained in bits 6-15 of the Transfer Frame Primary Header.
- c. The Spacecraft Identifier shall provide the identification of the destination spacecraft for the TC Transfer Frame.
- d. The Spacecraft Identifier shall be static throughout all mission phases.

NOTE 1 The Secretariat of the CCSDS assigns Spacecraft Identifiers according to the procedures in CCSDS 320.0-B-4.

NOTE 2 Different Spacecraft Identifiers can be assigned for normal operations and for development vehicles using the ground networks during pre-launch test operations, and for simulated data streams.

NOTE 3 A single physical channel can carry virtual channels with different Spacecraft Identifiers. Therefore the Spacecraft Identifier forms part of the full identification of a virtual channel within the physical channel.

6.2.2.7 Virtual Channel Identifier

- a. The Virtual Channel Identifier shall always be present in a Transfer Frame Primary Header.
- b. The Virtual Channel Identifier shall be contained within bits 16-21 of the Transfer Frame Primary Header.
- c. The Virtual Channel Identifier, together with the Spacecraft Identifier, shall identify the virtual channel to which the Transfer Frame belongs.

NOTE 1 The Virtual Channel Identifier is also used in the CLCW defined in clause 6.3. As described in clause 6.3.6, the choice of values for the identifiers of the virtual channels can affect the risk of misidentifying the virtual channel for a CLCW.

NOTE 2 The Spacecraft Identifier forms part of the identification of a virtual channel within a physical channel.

6.2.2.8 Frame Length

- a. The Frame Length shall always be present in a Transfer Frame Primary Header.
- b. The Frame Length shall be contained in bits 22-31 of the Transfer Frame Primary Header.
- c. The Frame Length shall contain a binary number equal to the number of octets in the TC Transfer Frame minus 1.

NOTE The value contained in the Frame Length is in the range 7 to 1023, corresponding to TC Transfer Frame lengths of 8 to 1024 octets.

- d. The value contained in the Frame Length may be variable.

NOTE Earlier definitions limited the TC Transfer Frame length to a maximum of 256 octets. In the earlier definitions, the Frame Length is contained in bits 24-31 of the Transfer Frame Primary Header, and is preceded by a 2-bit reserved field containing zeroes. Therefore, frames that conform to the earlier definition are valid under the current definition.

6.2.2.9 Frame Sequence Number

- a. The Frame Sequence Number shall always be present in a Transfer Frame Primary Header.
- b. The Frame Sequence Number shall be contained in bits 32-39 of the Transfer Frame Primary Header.
- c. In type-A Transfer Frames the Frame Sequence Number shall contain the COP-1 value N(S).

NOTE 1 Clause 7.9.2.6 specifies the setting of the COP-1 value N(S).

NOTE 2 The value is used in the COP-1 procedures at the receiving end to check that incoming type-A Transfer Frames are in sequence.

- d. In type-B Transfer Frames the Frame Sequence Number shall be set to zero.

NOTE The Bypass Flag defined in clause 6.2.2.3 indicates if a TC Transfer Frame is type-A or type-B.

6.2.3 Transfer Frame Data Field

- a. The Transfer Frame Data Field shall always be present in a TC Transfer Frame.
- b. The Transfer Frame Data Field shall follow, without gap, the Transfer Frame Primary Header.
- c. The length of the Transfer Frame Data Field shall be an integral number of octets.

NOTE 1 The length of the Transfer Frame Data Field is constrained by the length of the total TC Transfer Frame. The constraints on the length of a TC Transfer Frame are described in clause 6.2.1.

NOTE 2 The length of the Transfer Frame Data Field is not less than 1 octet and does not exceed 1017 octets.

- d. In type-D Transfer Frames the Transfer Frame Data Field shall contain a frame data unit.

NOTE The Transfer Frame Data Field of a type-D Transfer Frame contains exactly one frame data unit. The length of a frame data unit is therefore constrained to be a valid length for the field.

- e. In type-C Transfer Frames the Transfer Frame Data Field shall contain a COP-1 control command.

NOTE The Control Command Flag defined in clause 6.2.2.3 indicates if a TC Transfer Frame is type-D or type-C. Clause 6.1.1 describes the frame data unit. The COP-1 control commands are specified in clause 7.8.

6.2.4 Frame Error Control Field

6.2.4.1 General

- a. The Frame Error Control Field shall always be present in a TC Transfer Frame.
- b. The Frame Error Control Field shall follow, without gap, the Transfer Frame Data Field.
- c. The Frame Error Control Field shall have a length of two octets.
- d. The error detection encoding and decoding procedures specified in clauses 6.2.4.2 and 6.2.4.3 shall be used.

NOTE 1 This field provides a capability for detecting errors introduced into the frame during the transmission and data handling process. One reason for including this field in all TC Transfer Frames is to protect against residual errors after codeblocks are decoded in error-correcting mode as specified in clause 8.8.2.

NOTE 2 When applied to an encoded block of less than 32 768 bits (4096 octets), the code has the following capabilities:

- All error sequences composed of an odd number of bit errors are detected.
- All error sequences containing two-bit errors anywhere in the encoded block are detected.
- If a random error sequence containing an even number of bit errors (≥ 4) occurs within the block, the probability that the error is undetected is approximately 2^{-15} (or 3×10^{-5}).
- All single error bursts spanning 16 bits or less are detected provided no other errors occur within the block.

6.2.4.2 Encoding Procedure

- a. The encoding procedure shall be as follows:
 1. The encoding procedure accepts an $(n-16)$ -bit TC Transfer Frame, excluding the Frame Error Control Field, and generates a systematic binary $(n, n-16)$ block code by appending a 16-bit Frame Error Control Field as the final 16 bits of the codeblock.
 2. The equation for the contents of the Frame Error Control Field is:

$$\text{FECF} = [(X^{16} \cdot M(X)) + (X^{(n-16)} \cdot L(X))] \text{ modulo } G(X)$$

where

- all arithmetic is modulo 2;
- FECF is the 16-bit Frame Error Control Field and the first bit transferred is the most significant bit, taken as the coefficient of the highest power of X ;
- n is the number of bits in the encoded message;
- $M(X)$ is the $(n-16)$ -bit information message to be encoded expressed as a polynomial with binary coefficients, and the first bit transferred is the most significant bit M_0 taken as the coefficient of the highest power of X ;
- $L(X)$ is the presetting polynomial given by:
- $L(X) = \text{Error!}$
- $G(X)$ is the generating polynomial given by:
- $G(X) = X^{16} + X^{12} + X^5 + 1$

NOTE 1 The $X^{(n-16)} \cdot L(X)$ term has the effect of presetting the shift register to all '1' state prior to encoding.

NOTE 2 The encoding has the characteristics of a cyclic redundancy code.

NOTE 3 A sample implementation of an encoder is described in Annex A.

6.2.4.3 Decoding Procedure

- a. The decoding procedure shall use an error detection syndrome, $S(X)$, given by:

$$S(X) = [(X^{16} \cdot C^*(X)) + (X^n \cdot L(X))] \text{ modulo } G(X)$$

where

- $C^*(X)$ is the received block, including the Frame Error Control Field, in polynomial form, with the first bit transferred being the most significant bit C_0 taken as the coefficient of the highest power of X , and
- $S(X)$ is the syndrome polynomial, which is zero if no error is detected and non-zero if an error is detected, with the most significant bit S_0 taken as the coefficient of the highest power of X .

NOTE A sample implementation of a decoder is described in Annex A.

- b. The Frame Error Control Field shall not be used for error correction.

NOTE The code is intended for error detection purposes only.