



**VHF air-ground and air-air Digital Link (VDL)
Mode 4 radio equipment;
Technical characteristics and methods of measurement
for aeronautical mobile (airborne) equipment;
Part 4: Point-to-point functions**

Reference

REN/AERO-00019-4

Keywords

aeronautical, digital, radio, testing, VHF

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:

<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:

<https://portal.etsi.org/People/CommitteeSupportStaff.aspx>

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2015.

All rights reserved.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are Trade Marks of ETSI registered for the benefit of its Members.
3GPP™ and **LTE™** are Trade Marks of ETSI registered for the benefit of its Members and
of the 3GPP Organizational Partners.
GSM® and the GSM logo are Trade Marks registered and owned by the GSM Association.

Contents

| | |
|--|----|
| Intellectual Property Rights | 6 |
| Foreword..... | 6 |
| Modal verbs terminology..... | 6 |
| Introduction | 7 |
| 1 Scope | 8 |
| 2 References | 9 |
| 2.1 Normative references | 9 |
| 2.2 Informative references..... | 9 |
| 3 Definitions and abbreviations..... | 10 |
| 3.1 Definitions..... | 10 |
| 3.1.1 Basic reference model definitions..... | 10 |
| 3.1.2 Service conventions definitions | 10 |
| 3.1.3 General definitions..... | 10 |
| 3.1.4 Definition of bit order | 10 |
| 3.2 Abbreviations | 10 |
| 4 General description of VDL Mode 4 mobile station point-to-point services | 12 |
| 4.1 General | 12 |
| 4.2 Data Link Service (DLS) and Link Management Entity (LME) | 12 |
| 4.2.1 General..... | 12 |
| 4.2.2 DLS timers..... | 15 |
| 4.2.3 DLS counters | 16 |
| 4.2.4 Toggle bit (T)..... | 16 |
| 4.2.5 State variables (T_i and T_r)..... | 16 |
| 4.2.6 Initialize Bit (IB)..... | 16 |
| 4.2.7 More bit (M) | 17 |
| 4.2.8 Priority (pr) | 17 |
| 4.2.9 Message fragmentation and concatenation | 17 |
| 4.2.10 DLS procedures | 17 |
| 4.2.10.1 Selection of DLS procedures..... | 17 |
| 4.2.10.2 Short transmission procedures | 17 |
| 4.2.10.3 Long transmission procedures..... | 18 |
| 4.2.11 Linking transmissions | 20 |
| 4.2.11.1 Types of linking | 20 |
| 4.2.11.2 Combined RTS / DATA DLPDUs..... | 20 |
| 4.2.11.3 Combined DATA / ACK DLPDUs..... | 20 |
| 4.2.11.4 Combined RTS / ACK DLPDUs | 20 |
| 4.2.11.5 Combined CTS / ACK DLPDUs | 20 |
| 4.2.12 Ground-air link management | 20 |
| 4.2.13 Air-air link management..... | 21 |
| 4.3 Additional VSS services..... | 21 |
| 5 Minimum performance specification under standard test conditions..... | 21 |
| 5.1 DLS sublayer..... | 21 |
| 5.1.1 General..... | 21 |
| 5.1.1.1 Services | 21 |
| 5.1.1.2 Data transfer | 22 |
| 5.1.1.3 DATA DLPDU duplicate suppression and sequencing | 22 |
| 5.1.1.4 Error detection..... | 22 |
| 5.1.1.5 Station identification | 22 |
| 5.1.1.6 Broadcast addressing..... | 22 |
| 5.1.1.7 DLS Priority..... | 23 |
| 5.1.1.8 DLS Link control DLPDUs | 23 |
| 5.1.2 DLS protocol specification | 23 |
| 5.1.2.1 State variables | 23 |
| 5.1.2.2 DLS burst formats | 23 |

| | | |
|----------|--|----|
| 5.1.3 | DLS system parameters | 27 |
| 5.1.4 | DLS procedures | 29 |
| 5.1.4.1 | Setting of re-transmission parameter | 29 |
| 5.1.4.2 | Selection of user data packet for transmission | 29 |
| 5.1.4.3 | Selection of transmission procedures | 31 |
| 5.1.4.4 | Short transmission procedures | 31 |
| 5.1.4.5 | Long transmission procedures | 32 |
| 5.1.4.6 | No link with sender | 34 |
| 5.1.4.7 | User data packet reception | 35 |
| 5.1.4.8 | Receipt of ACK DLPDU | 35 |
| 5.1.4.9 | Link reset | 36 |
| 5.1.4.10 | Linking DLS DLPDU transmissions | 37 |
| 5.1.4.11 | CTRL DLPDU | 38 |
| 5.1.4.12 | Procedures for air-air communication | 38 |
| 5.2 | Link management entity sublayer | 38 |
| 5.2.1 | Services | 38 |
| 5.2.1.1 | General | 38 |
| 5.2.1.2 | Link provision | 39 |
| 5.2.1.3 | Link change notifications | 39 |
| 5.2.1.4 | CTRL DLPDU | 39 |
| 5.2.1.5 | Broadcast link management burst | 40 |
| 5.2.2 | Control (CTRL) parameter formats | 40 |
| 5.2.2.1 | Encoding | 40 |
| 5.2.2.2 | General purpose information parameters | 40 |
| 5.2.2.3 | Mobile-initiated information parameters | 44 |
| 5.2.2.4 | Ground-initiated modification parameters | 45 |
| 5.2.2.5 | Ground-initiated information parameters | 48 |
| 5.2.3 | LME timers and parameters | 50 |
| 5.2.3.1 | General | 50 |
| 5.2.3.2 | Timer TL1 (maximum link overlap time) | 51 |
| 5.2.3.3 | Parameters TL2 (link initialization time) | 51 |
| 5.2.3.4 | Timer TL4 (leave generation latency) | 51 |
| 5.2.4 | CTRL DLPDU types and procedures | 51 |
| 5.2.5 | CTRL transmission procedures | 55 |
| 5.2.5a | Frequency management procedures | 55 |
| 5.2.5.1 | Link connectivity procedures | 55 |
| 5.2.5.2 | Ground Station Identification | 55 |
| 5.2.5.3 | Link establishment | 56 |
| 5.2.5.4 | Mobile-initiated handoff | 57 |
| 5.2.5.5 | Mobile-requested ground-initiated handoff | 58 |
| 5.2.5.6 | Ground-initiated handoff | 58 |
| 5.2.5.7 | Ground-requested mobile-initiated handoff | 59 |
| 5.2.5.8 | Ground-requested broadcast handoff | 59 |
| 5.2.5.9 | Ground-commanded autotune | 60 |
| 5.2.6 | VDL Mode 4 Mobile SubNetwork Dependent Convergence Function (SND CF) | 60 |
| 5.2.6.1 | Frame mode SND CF | 60 |
| 5.3 | Additional VSS requirements | 60 |
| 5.3.1 | Information transfer request protocol specification | 60 |
| 5.3.1.1 | Information transfer request parameters | 60 |
| 5.3.1.2 | Information transfer request transmission procedures | 62 |
| 5.3.1.3 | Information transfer request acknowledgement procedures | 62 |
| 5.3.2 | Void | 62 |
| 5.3.3 | Void | 62 |
| 5.3.4 | Retransmission procedures | 63 |
| 6 | General design requirements | 63 |
| 7 | Protocol test procedures | 63 |
| 7.1 | General | 63 |
| 7.2 | Required test rig | 63 |
| 7.3 | Protocol test-suite description methodology | 64 |
| 7.4 | Detailed protocol test procedures | 64 |

| | | |
|--|---|------------|
| 7.4.1 | Test-suite overview | 64 |
| 7.4.2 | Declarations | 67 |
| 7.4.3 | Constraints | 67 |
| 7.4.3.1 | Abbreviations | 67 |
| 7.4.3.1.1 | Subfield mnemonics | 67 |
| 7.4.3.1.2 | Special characters used in the subfield definitions | 68 |
| 7.4.3.1.3 | Station addresses and positions | 68 |
| 7.4.3.1.4 | VDL bursts | 69 |
| 7.4.3.2 | Test cases | 95 |
| 7.4.3.2.1 | Test case macros | 96 |
| 7.4.3.2.2 | Test case descriptions | 98 |
| Annex A (informative): Cross reference matrix | | 172 |
| History | | 185 |

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<http://ipr.etsi.org>).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This European Standard (EN) has been produced by ETSI Technical Committee Aeronautics (AERO).

The present document is part 4 of a multi-part deliverable covering the VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment, as identified below:

- Part 1: "Physical layer";
- Part 2: "General description and data link layer";
- Part 3: "Additional broadcast aspects";
- Part 4: "Point-to-point functions".**

The present document is accompanied by an equivalent ground-based standard, ETSI EN 301 842 [i.1] parts 1 to 5, covering the VHF air-ground Data Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for ground-based equipment.

NOTE: Minimum Operational Performance Specifications (MOPS) are also being developed for VDL Mode 4. EUROCAE have previously published Interim MOPS for VDL Mode 4 [i.2] which are a sub set of ETSI EN 302 842-1 [5], ETSI EN 302 842-2 [6], ETSI EN 302 842-3 [7] and ETSI EN 302 842-4 (the present document). ETSI EN 302 842-1 [5], ETSI EN 302 842-2 [6], ETSI EN 302 842-3 [7] and ETSI EN 302 842-4 (the present document) comply with the requirements of EC Mandate M/318 [i.3].

| National transposition dates | |
|--|-----------------|
| Date of adoption of this EN: | 6 April 2015 |
| Date of latest announcement of this EN (doa): | 31 July 2015 |
| Date of latest publication of new National Standard or endorsement of this EN (dop/e): | 31 January 2016 |
| Date of withdrawal of any conflicting National Standard (dow): | 31 January 2017 |

Modal verbs terminology

In the present document **"shall"**, **"shall not"**, **"should"**, **"should not"**, **"may"**, **"need not"**, **"will"**, **"will not"**, **"can"** and **"cannot"** are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"must" and **"must not"** are **NOT** allowed in ETSI deliverables except when used in direct citation.

Introduction

The present document states the technical specifications for Very High Frequency (VHF) Digital Link (VDL) Mode 4 aeronautical mobile (airborne) radio transmitters, transceivers and receivers for air-ground and air-air communications operating in the VHF band, using Gaussian Filtered Frequency Shift Keying (GFSK) modulation with 25 kHz channel spacing and capable of tuning to any of the 25 kHz channels from 112,000 MHz to 136,975 MHz as defined in ICAO VDL SARPs [2].

The present document may be used to produce tests for the assessment of the performance of the equipment. The performance of the equipment submitted for type testing should be representative of the performance of the corresponding production model.

The present document has been written on the assumption that:

- the type test measurements will be performed only once, in an accredited test laboratory and the measurements accepted by the various authorities in order to grant type approval;
- if equipment available on the market is required to be checked it will be tested in accordance with the methods of measurement specified in the present document or a documented alternative approved by the certifying authority;
- equipment comply with ETSI EN 302 842-1 [5] and ETSI EN 302 842-2 [6].

NOTE: The present document has been produced with a view to maintaining consistency of numbering with the equivalent standard for ground equipment (ETSI EN 301 842 [i.1]). Where requirements are the same, they have been given the same number. Some new airborne requirements have been inserted between requirements that were sequential in ETSI EN 301 842 [i.1]. This has led to a non-standard form of numbering for new requirements in some places.

1 Scope

The present document applies to the following radio equipment types:

- Very High Frequency (VHF) Digital Link (VDL) Mode 4 aeronautical mobile (airborne) radio transmitters, transceivers and receivers for air-ground and air-air communications operating in the VHF band, using Gaussian Filtered Frequency Shift Keying (GFSK) modulation with 25 kHz channel spacing and capable of tuning to any of the 25 kHz channels from 112,000 MHz to 136,975 MHz as defined in ICAO VDL SARPs [2].

The present document provides part 4 of the technical specifications.

The present document is designed to ensure that equipment certified to it will be compatible with the relevant ICAO VDL SARPs [2] and ICAO VDL4 Technical Manual [1].

Manufacturers should note that in future the tuning range for the transmitter and receiver may also cover any 25 kHz channel from 108,000 MHz to 111,975 MHz.

The present document applies to "aeronautical mobile (airborne and as applicable also ground vehicles)" equipment which will hereinafter be referred to as "mobile" equipment.

The scope of the present document is limited to mobile stations. The equivalent specification for ground stations is ETSI EN 301 842 [i.1].

A description of the scope of the VDL Mode 4 system is provided in part 2 of ETSI EN 302 842-2 [6], clause 1.

ETSI EN 302 842-1 [5] deals with tests of the physical layer, ETSI EN 302 842-2 [6] deals with core link layer functionality and ETSI EN 302 842-3 [7] with additional broadcast functionality. The present document deals with tests of the datalink layer (DLS) and the link layer sufficient to support point-to-point functionality. Note that, as described in ETSI EN 302 842-2 [6], a system supporting point-to-point functionality is required to conform to ETSI EN 302 842-1 [5], ETSI EN 302 842-2 [6] and the specification in the present document.

The present document is organized as follows:

- clause 2 provides references to relevant documents;
- clause 3 provides general definitions, abbreviations and symbols used;
- clause 4 describes the VDL Mode 4 mobile station point-to-point functions;
- clause 5 provides performance specifications for the VDL Mode 4 mobile station;
- clause 6 provides general design requirements;
- clause 7 provides protocol tests for the point-to-point functions of the system;
- annex A provides a detailed cross-reference to the relevant requirements contained in ICAO VDL4 Technical Manual [1].

Note that the system can support a very wide range of functions. It is not practical to provide specific tests for all aspects of functionality. The approach used is to provide detailed tests for the core point-to-point functionality and to provide tests of those remaining requirements which, if wrongly implemented, could cause a deterioration in the service offered by other VDL Mode 4 stations.

Mandating and Recommendation Phrases

- a) "Shall":
 - The use of the word "Shall" indicates a mandated criterion; i.e. compliance with the particular procedure or specification is mandatory and no alternative may be applied.

b) "Should":

- The use of the word "Should" (and phrases such as "It is recommended that...", etc.) indicates that though the procedure or criterion is regarded as the preferred option, alternative procedures, specifications or criteria may be applied, provided that the manufacturer, installer or tester can provide information or data to adequately support and justify the alternative.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at <http://docbox.etsi.org/Reference>.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

- [1] ICAO Doc 9816 (First Edition 2004): "Manual on VHF Digital Link (VDL) Mode 4 - Part 2: Detailed Technical Specifications".
- [2] ICAO Annex 10 to the Convention on International Civil Aviation: "Aeronautical Telecommunications, Volume III: Communication Systems, Part I: Digital Data Communication Systems, chapter 6", including Amendment 88 (applicable 14/11/13)".
- [3] ICAO 9705-CD: "Manual of Technical Provisions for the Aeronautical Telecommunication Network (ATN)".
- [4] Void.
- [5] ETSI EN 302 842-1 (V1.3.1): "VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment; Part 1: Physical layer".
- [6] ETSI EN 302 842-2 (V1.4.1): "VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment; Part 2: General description and data link layer".
- [7] ETSI EN 302 842-3 (V1.4.1): "VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment; Part 3: Additional broadcast aspects".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

- [i.1] ETSI EN 301 842 (all parts): "VHF air-ground Data Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for ground-based equipment".
- [i.2] EUROCAE ED-108A: "MOPS for VDL Mode 4 Aircraft Transceiver for ADS-B", July 2001.

[i.3] EC Mandate M/318: "Mandate to CEN/CENELEC/ETSI for standardisation in the field of air traffic management systems and Galileo local components".

3 Definitions and abbreviations

3.1 Definitions

3.1.1 Basic reference model definitions

See ETSI EN 302 842-2 [6], clause 3.1.1.

3.1.2 Service conventions definitions

See ETSI EN 302 842-2 [6], clause 3.1.2.

3.1.3 General definitions

For the purposes of the present document, the terms and definitions given in ETSI EN 302 842-1 [5] clause 3.1.3, ETSI EN 302 842-2 [6] clause 3.1.3 and the following apply:

long transmission procedure: procedure that allows transfer of DATA packets in reserved slots, avoiding the loss rates and delays associated with random access

NOTE: This includes the ability to link sequences of DATA packets, providing continuous transfer in reserved slots.

More bit (M): bit that controls message fragmentation and concatenation

NOTE: It is set to zero to indicate the end of a message. It is set to 1 to indicate that it is part of a fragmented message and that there are more fragments to follow.

Network Setup Connection Oriented Protocol (NSCOP): protocol that enables a ground-air link between mobiles

priority (pr): priority of a message, or, in the case of an RTS, the priority of the requested data

short transmission procedure: procedure for the sending of a single data packet. It contains a reservation for a subsequent acknowledgement

Toggle bit (T): bit that allows detection and rejection of duplicate DLPDUs

NOTE: The T bit is alternately set to 1 and 0 on each successive DATA transmission, except for retransmissions.

Zero-Overhead Connection-Oriented Protocol (ZOCOP): protocol that enables an air-air link between mobiles

3.1.4 Definition of bit order

In the tables included in the present document to illustrate the format of bursts, the following order is implied:

- a) bit order in each burst subfield are indicated by subscript numbers. Bit 1 indicates the least significant bit; and
- b) bits are transmitted octet by octet, starting with the first octet in each table, and within each octet the rightmost bit (as shown in the tables) is transmitted first.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

| | |
|-------|--|
| ACK | ACKnowledgement (burst) |
| ADM | ADMinistration identifier |
| ADS-B | Automatic Dependent Surveillance-Broadcast |
| ARS | Administration Region Selector |
| ATN | Aeronautical Telecommunication Network |
| CMD | CoMmand |

| | |
|--------|--|
| CPR | Compact Position Reporting |
| CRC | Cyclic Redundancy Code |
| CTRL | ConTRoL (DLPDU) |
| CTS | Clear To Send (burst) |
| DISC | DISConnect |
| DLE | Data Link Entity |
| DLPDU | Data Link Protocol Data Unit |
| DLS | Data Link Service |
| DM | Disconnected Mode (DLPDU) |
| DTE | Data Terminal Equipment |
| EC | European Commission |
| erid | extended reservation ID |
| FRMR | FRaMe Reject (DLPDU) |
| GFSK | Gaussian Filtered Frequency Shift Keying |
| GSC | Global Signalling Channel |
| GSIF | Ground Station Information Frame |
| hex | hexadecimal |
| HO | Hand Off |
| IA | International Alphabet |
| IA-5 | International Alphabet - 5 |
| IB | Initialize Bit |
| ICAO | International Civil Aviation Organization |
| ID | IDentity |
| INFO | INFormation (DLPDU) |
| ISO | International Organization for Standardization |
| IS-SME | Intermediate System - System Management Entity |
| LCR | Link Connection Refused |
| LE | Link Establishment |
| lg | length |
| LME | Link Management Entity |
| M | More bit |
| MAC | Media Access Control |
| MOPS | Minimum Operational Performance Specification |
| NACK | Negative ACKnowledgement |
| neg | negotiation |
| NETs | Network Entity Titles |
| NM | Nautical Mile |
| NSCOP | Network Setup Connection Orientated Protocol |
| O | Optional |
| p | priority |
| PCO | Point of Control and Observation |
| PI | Parameter Identifier |
| pr | priority |
| QoS | Quality of Service |
| Res | Reserved header bits |
| RF | Radio Frequency |
| rid | reservation ID |
| RSP | ReSPonse |
| RTS | Request To Send (DLPDU) |
| SARPs | Standards And Recommended Practices |
| seq | sequence |
| SNDCF | SubNetwork Dependent Convergence Function |
| SZOM | Start Zero Overhead Mode (DLPDU) |
| T | Toggle bit |
| UCTRL | Unacknowledged ConTRoL data broadcast (DLPDU) |
| UDATA | Unacknowledged DATA broadcast (DLPDU) |
| UINFO | Unacknowledged user INFormation data broadcast (DLPDU) |
| VDL | VHF Digital Link |
| VHF | Very High Frequency |
| VME | VDL Management Entity |
| VSS | VDL Mode 4 Specific Services |
| ZOCOP | Zero-Overhead Connection-Oriented Protocol |

4 General description of VDL Mode 4 mobile station point-to-point services

4.1 General

A description of VDL Mode 4 is provided in ETSI EN 302 842-2 [6]. This clause provides a description of the datalink layer and associated services of the VSS and LME required to support point-to-point communications.

The specifications in the present document provide air-to-ground and air-to-air point-to-point services based on the ICAO Technical Manual. The present document includes:

- Point-to-point data and control data transfer functions from the DLS.
- Point-to-point link control within the LME.

Note that the present document covers the mobile response to the establishment, termination and handover of links between ground stations. Decisions made by ground stations to establish, terminate or handover links between ground stations are local issues and beyond the scope of these specifications.

In most respects, the VDL Mode 4 mobile station follows the provisions of the ICAO standards material for VDL Mode 4. Within the ICAO standard, there are some requirements which apply explicitly only to ground stations. Requirements on ground stations are covered in ETSI EN 301 842 (all parts) [i.1].

The scope of the present document is for an mobile station supporting point-to-point applications. Hence the ability to support broadcast communication is not included in the present document.

Note that, although certain protocols will not be used by the mobile station, the ability to recognize the use by ground stations of these protocols and to respond in a consistent manner is an mobile station requirement and is included in the present document.

4.2 Data Link Service (DLS) and Link Management Entity (LME)

4.2.1 General

The Data Link Service (DLS) provides a point-to-point protocol supporting both air-to-ground and air-to-air exchanges. For air-to-ground exchanges, connection management is handled by the LME, using a Negotiated Setup Connection-Orientated Protocol (NSCOP). For air-to-air exchanges, the DLS defines a ZOCOP protocol with link establishment and disestablishment controlled by timers, allowing rapid link negotiation between aircraft.

The specifications in the present document assume that VDL Mode 4 will be used with a network based on ATN protocols (see ICAO ATN SARPs [3]). It should be noted that, in future, it could support other network protocols.

The DLS is a simple information exchange protocol which provides:

- a) Explicit acknowledgement of each DATA packet.
- b) Mechanisms to provide data transfer in reserved slots, avoiding loss rates and delays associated with random access. This includes the ability to link sequences of DATA packets, providing continuous transfer in reserved slots.
- c) Packet fragmentation, to allow long user data packets to be transferred across the link in fragments optimally matched to the link conditions.
- d) Duplicate detection and suppression via a simple toggle-bit mechanism.
- e) Explicit support for the 15 priority levels defined for the ATN, including the ability for high priority messages to be sent in the middle of a fragmented low priority message.

As illustrated in figure 4.1, the DLS:

- Provides services to external users in order to support point-to-point, multi-cast and broadcast communications.
- Provides services to the LME to support link management.
- Supports services between peer DLSs.
- Uses the services of the VSS in order to send and receive messages.

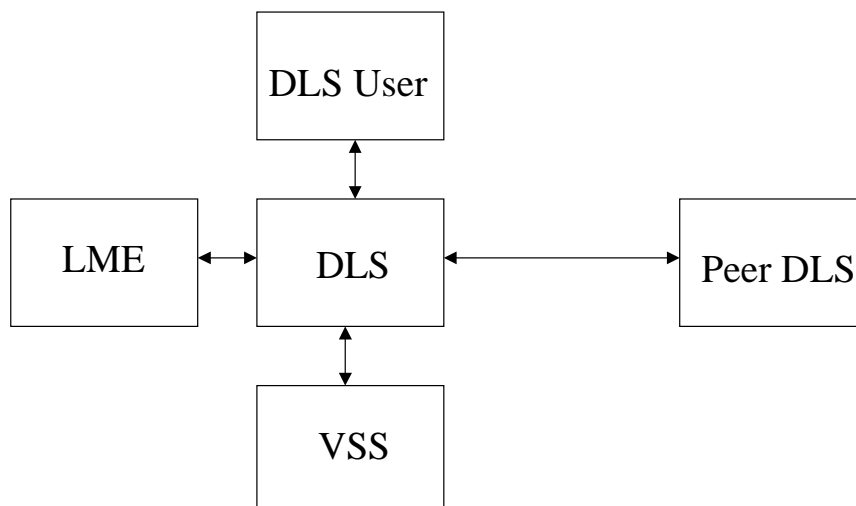


Figure 4.1: Functions of the DLS

Figure 4.2 illustrates the use of various services provided by the DLS and VSS in supporting the DLS User and the LME.

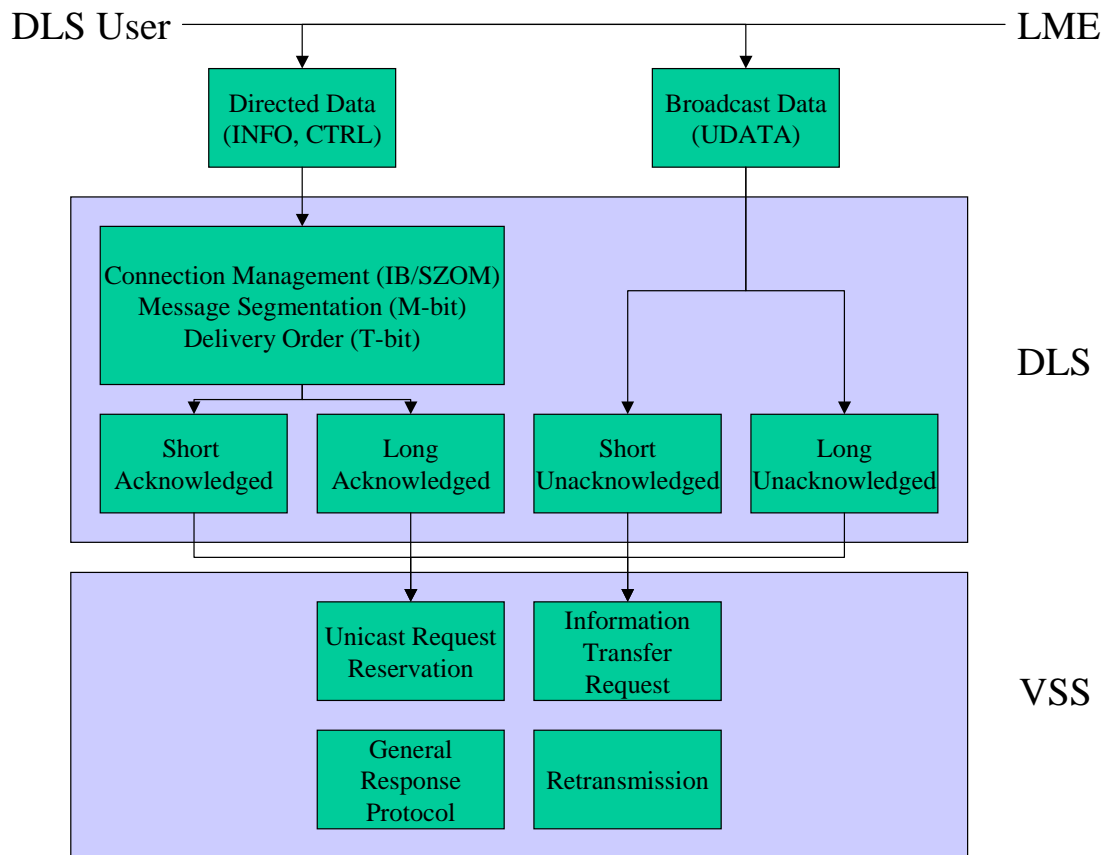


Figure 4.2: Services provided by the DLS and VSS

Protocols for short unacknowledged broadcast data have already been specified in the core link layer functionality (ETSI EN 302 842-2 [6]).

A DLS Burst consists of:

- Burst ID (fixed or variable).
- One or two DLPDUs.

The following DATA DLPDUs (data link protocol data units) are defined:

- CTRL: Used by the LME to establish and maintain links for NSCOP communication.
- INFO: Contains a user data field.
- UDATA: Either a UCTRL or a UINFO. Enables the broadcast of user data - it is unacknowledged.

The following RTS DLPDUs are defined:

- CTRL_RTS: Sent to request the sending of link maintenance data.
- INFO_RTS: A request to send user data (for long transmission procedures).
- UDATA_RTS: A request to send broadcast data.

The following ACK, CTS and other DLS link control DLPDUs are defined:

- UDATA_CTS: This DLPDU is a clear to send in response to a UDATA_RTS.
- INFO_ACK: An acknowledgement of the previous INFO DLPDU.
- INFO_CTS: A clear to send (for long transmission procedures).

- CTRL_ACK: An acknowledgement of the previous CTRL DLPDU.
- FRMR: Used to reset the link.
- FRMR_ACK: Sent to acknowledge receipt of FRMR.
- DM/DISC (Disconnected Mode): Used to indicate that a DLS DLPDU has been received when no link has been initialized.
- DM/FRMR - Used to indicate that a DLS DLPDU has been received when the link is in the process of being initialized.
- SZOM: sent in combination with an INFO, INFO_RTS or INFO_ACK to establish a link for ZOCOP communication.

Table 4.1 shows the DLS burst types.

Table 4.1: DLPDU contents

| | Toggle Bit (T) | More Bit (M) | Priority (pr) | Negotiation subfield (neg) | Initialize bit IB | Length (lg) |
|-----------|----------------|--------------|---------------|----------------------------|-------------------|-------------|
| CTRL | ✓ | ✓ | | | | |
| INFO | ✓ | ✓ | ✓ | | | |
| UDATA | | | | | | |
| CTRL_RTS | ✓ | | | | ✓ | ✓ |
| INFO_RTS | ✓ | | ✓ | | | ✓ |
| UDATA_RTS | | | ✓ | | | ✓ |
| UDATA_CTS | | | | | | |
| INFO_ACK | ✓ | | | | | |
| INFO_CTS | | | | | | |
| CTRL_ACK | ✓ | | | | | |
| CTRL_CTS | | | | | | |
| FRMR_ACK | | | | | | |
| FRMR | | | | | | |
| DM/DISC | | | | | | |
| DM/FRMR | | | | | | |
| SZOM | | | | ✓ | | |

Where:

- Toggle Bit (T): This is used for duplicate detection and rejection.
- More Bit (M): This is set to zero to indicate the end of a message. It is set to 1 to indicate that it is part of a fragmented message and that there are more fragments to follow.
- priority (pr): This is the priority of the message.
- negotiation subfield (neg): This indicates the link management parameters to be used for air-to-air link control.
- Initialize Bit (IB): This bit causes the receiver to initialize the Tt and Tr state variables and to clear the send and receive arrays whilst processing the burst. It is set to zero and ignored on receipt unless otherwise stated.
- length (lg): This is the length of the data message in slots.

4.2.2 DLS timers

The following timers are used in the DLS procedures:

- TD1: ZOCOP link transmit timer. This timer is reset when a burst is sent to the peer. If it expires, then Tt is set to 0 and the send channel array cleared.
- TD2: ZOCOP link receive reset timer. This timer is reset when a burst is received from the peer. If this timer expires then Tt is set to 0, the send channel array is cleared and the link is considered to be terminated.

4.2.3 DLS counters

The following counters are used in the DLS procedures:

- ND1: Maximum number of octets that may be submitted to the DLS for transfer.
- ND2: Maximum length of a DLS transmission that may use the short transmission procedures.
- ND3: Maximum length of a fragment in slots with M-bit processing.
- ND4: Maximum length of UDATA burst.

4.2.4 Toggle bit (T)

The Toggle (T) bit allows detection and rejection of duplicate DLPDUs. The T bit is alternately set to 1 and 0 on each successive DATA transmission, except for retransmissions. The associated RTS, CTS and ACK DLPDUs should have the same value T bit as the DATA packet.

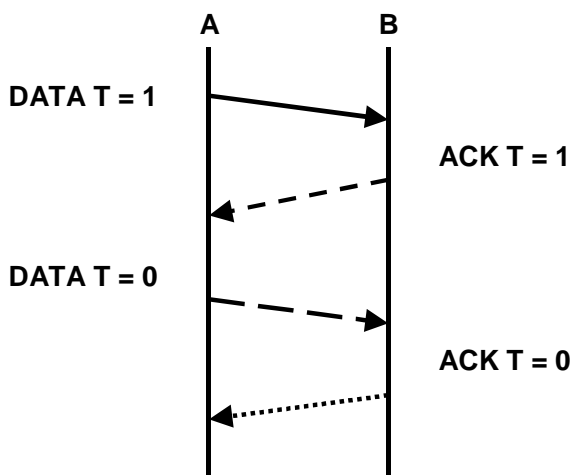


Figure 4.3: Use of the T bit in a short procedure

4.2.5 State variables (T_t and T_r)

When a message being transmitted is not the first message between the two stations, the sending station sets the T bit to the inverse value that it had for the previous message. The value it transmitted previously is stored in the state variable T_t .

The receiving station compares the value of the T bit it receives with the value it received previously, to check that the received T bit is of the correct value. The value it received previously is stored in the state variable T_r .

When the sending station receives an ACK or a CTS, it compares the value of the T bit with the value it sent, which was stored in T_t , to check that it has the correct value.

Messages may be discarded or re-sent if the T bit is found to be incorrect.

4.2.6 Initialize Bit (IB)

When the first message to another station is being transmitted, the sending station always sets the T bit to zero, and correspondingly sets its value of T_t to zero. For the first message transmitted between two stations, the receiving station should not compare the T bit it receives with a previous value of T_r .

Therefore there is an Initialize Bit (IB) in the messages, which is usually set to zero, but which for the first message is always set to 1. When the receiving station receives a message with $IB = 1$ and $T = 0$, it knows to reset its value of T_r to zero instead of performing a comparison.

4.2.7 More bit (M)

The more bit (M) controls message fragmentation and concatenation. It is set to zero to indicate the end of a message. It is set to 1 to indicate that it is part of a fragmented message and that there are more fragments to follow.

4.2.8 Priority (pr)

The priority (pr) is the priority of the message. The priority levels are defined by the Q1 quality of service parameter.

4.2.9 Message fragmentation and concatenation

If the length of the burst is longer than ND3 slots, then the sending station will fragment the message.

For single fragment messages:

- The M bit is set to zero.

For multiple fragment messages:

- The M bit is set to 1, except the last fragment, which is set to zero.

If any preceding message fragments have been received with the M bit set to 1, then the user data part of the DATA DLPDU is concatenated to the end of the message fragments.

If the M bit is set to zero then the user data part of the DATA DLPDU, along with any other message fragments received earlier, is passed to the DLS user as a single message.

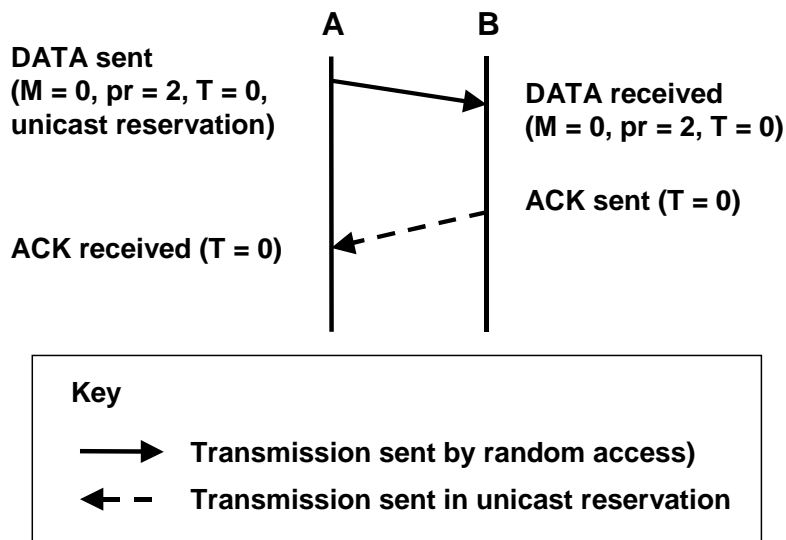
4.2.10 DLS procedures

4.2.10.1 Selection of DLS procedures

There are two types of DLS procedures, short and long. After receiving the burst (from the DLS user), the sending station calculates the total length of the transmission. If the total length is less than or equal to N3, then the station will use the short transmission procedures, otherwise, the station will use the long transmission procedures.

4.2.10.2 Short transmission procedures

The short transmission procedures involve the sending of a single data packet that contains a reservation for a subsequent acknowledgement.



NOTE: The values shown in the figure for pr and T are for the purpose of example.

Figure 4.4: Short transmission procedures

Transmission of DATA DLPDU

- The sending station transmits a burst containing a DATA DLPDU.
- The M bit is set to 0.
- The pr subfield is set to the priority of the DATA DLPDU.
- The T bit is set to 0 or 1 (inverse to previous transmission).
- The burst may include a CTRL or INFO DLPDU containing a unicast request reservation field for the acknowledgement.

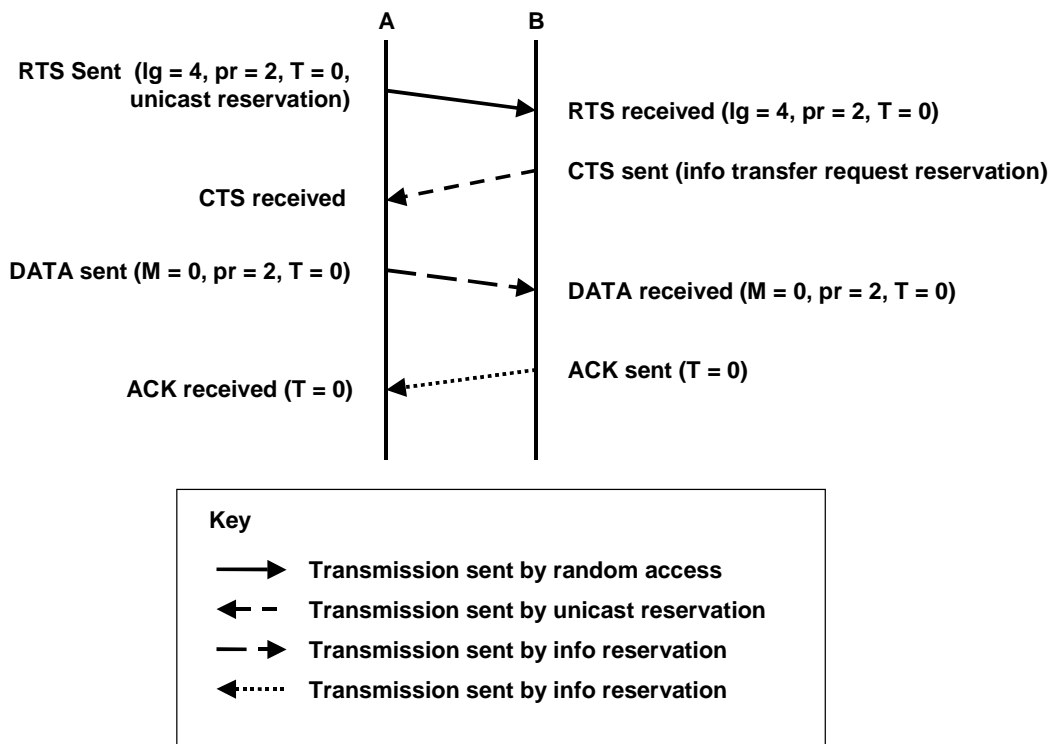
Transmission of ACK DLPDU

- The receiving station transmits a burst containing an ACK DLPDU in the slot reserved by the unicast request reservation in the received DATA DLPDU.
- The T bit is set to the value of the T bit in the received DATA DLPDU.

4.2.10.3 Long transmission procedures

The long transmission procedures allow transfer of DATA packets in reserved slots, avoiding the loss rates and delays associated with random access. This includes the ability to link sequences of DATA packets, providing continuous transfer in reserved slots.

The long transmission procedure starts with an Request-To-Send (RTS) being sent by random access. (The RTS is in general shorter than the DATA transmission so it is quicker to find an available slot for sending this by random access). The receiving station responds by sending a Clear-To-Send (CTS), together with an information transfer request reservation - this reserves slots for the DATA from the sending station and for the subsequent ACK.



NOTE: The values shown in the figure for pr and T are for the purpose of example.

Figure 4.5: Long transmission procedures

Transmission of RTS

- The sending station transmits a burst containing an RTS DLPDU.
- The pr subfield is set to the priority of the INFO_RTS or UDATA_RTS DLPDU to be transmitted.
- The lg subfield is set to the length of the DLS burst required to contain the DATA DLPDU.
- The T bit is set to zero or 1 (inverse to previous transmission).

Transmission of CTS

- The receiving station transmits a burst containing a CTS DLPDU in the slot reserved by the unicast request reservation in the received RTS DLPDU.
- The burst also contains an information transfer request reservation to reserve slots for the expected DATA transmission and the subsequent acknowledgement.

Transmission of DATA DLPDU

- The sending station transmits a burst containing a DATA DLPDU.
- The M bit is set to 0 if the data fragment is not to be followed by any others, or 1 if it is.
- The T bit is set to the T bit of the RTS.

Transmission of ACK DLPDU

- The receiving station transmits a burst containing an ACK DLPDU in the slot reserved by the unicast request reservation in the received DATA DLPDU.
- The T bit is set to the value of the T bit in the received DATA DLPDU.

4.2.11 Linking transmissions

4.2.11.1 Types of linking

A station with a queue of transmissions for the same receiving station can link them in the following ways:

- Combine an RTS DLPDU with a DATA DLPDU.
- Combine a CTS DLPDU with an ACK DLPDU.
- Combine a DATA DLPDU with an ACK DLPDU.
- Combine an RTS DLPDU with an ACK DLPDU.

4.2.11.2 Combined RTS / DATA DLPDUs

In the long transmission procedure, the sending station can send the DATA of the first message with an attached RTS for the DATA of the second message. The receiving station then sends an ACK for the first message along with a CTS for the second.

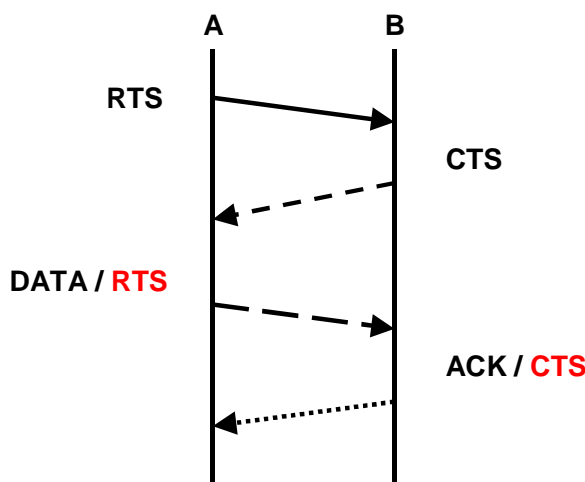


Figure 4.6: Linked transmissions

4.2.11.3 Combined DATA / ACK DLPDUs

If the receiving station has some data to send which would fit in one slot it can send it with the ACK transmission.

4.2.11.4 Combined RTS / ACK DLPDUs

A receiving station that has data to send that does not fit in one slot can send an RTS for itself with the ACK.

4.2.11.5 Combined CTS / ACK DLPDUs

A receiving station that has received data and needs to respond to an RTS can send a CTS for more data with the ACK for the received data.

4.2.12 Ground-air link management

For ground-air link management, the CTRL DLPDU is used by the LME to establish and maintain the link. The CTRL_CMD is sent to initiate a link and the CTRL_RSP is sent in response. On receiving a CTRL_RSP the link has been made.

4.2.13 Air-air link management

Air-air link management is supported by a Zero-Overhead Connection Oriented Protocol (ZOCOP), which makes use of the SZOM DLPDU. A mobile wishing to send data to another mobile it does not currently have a connection with should:

- combine an SZOM with the first data transmission, if using the short transmission procedure; or
- combine an SZOM with the first RTS transmission, if using the long transmission procedure.

On receiving an ACK or CTS after transmission of the SZOM, the air-air link is considered established, the timers are set, and data exchange is carried out as previously described. The link is considered terminated when the timers expire.

4.3 Additional VSS services

The correct operation of the DLS requires an additional service from the VSS, namely the information transfer request protocol.

5 Minimum performance specification under standard test conditions

5.1 DLS sublayer

5.1.1 General

5.1.1.1 Services

| Requirement reference | |
|---|--|
| 5.1.1.1.1 | The VDL Link Layer shall provide a reliable point-to-point service using a connection oriented DLS sublayer. |
| 5.1.1.1.2 | The VDL Link Layer shall provide an unacknowledged broadcast service using a connectionless DLS sublayer as defined in ETSI EN 302 842-2 [6], clause 5.3. |
| 5.1.1.1.3 | The DLS shall support communications on a shared communications channel as described in this clause. |
| 5.1.1.1.4 | The DLS shall support bit-orientated simplex communications using a negotiated setup connection-orientated protocol (NSCOP) and a zero-overhead connection-orientated protocol (ZOCOP) between DLE pairs. |
| 5.1.1.1.5 | The DLS shall provide the following services: <ol style="list-style-type: none"> transmission of user data; indication that user data has been sent; reception of user data; indication that DLS link has been established; indication that the DLS link has been broken. |
| 5.1.1.1.6 | Stations supporting the point-to-point communications functionality provided by the DLS shall simultaneously support at least 8 peer-to-peer links with other stations. |
| NOTE 1: It is intended that NSCOP be used for air/ground (A/G) communications. | |
| NOTE 2: Apart from the procedures for link set-up and tear down, the NSCOP and ZOCOP protocols operate identically. | |
| NOTE 3: Any two stations have one DLE pair per frequency. | |

5.1.1.2 Data transfer

| Requirement reference | |
|--|---|
| 5.1.1.2.1 | User data packets and LME data shall be transferred in the information fields of INFO, UDATA and CTRL data link protocol data units (DLPDUs) which are collectively known as DATA DLPDUs. |
| 5.1.1.2.2 | LME data shall be contained in CTRL and UCTRL frames only. |
| 5.1.1.2.3 | The link layer shall process the largest packet size, specified in clause 5.1.3.8, without fragmenting. |
| 5.1.1.2.4 | Larger packets shall be fragmented according to the procedures of clauses 5.1.4.2.5 to 5.1.4.2.14. |
| 5.1.1.2.5 | Only one data link user packet shall be contained in a DATA DLPDU. |
| NOTE 1: The Frame Mode Subnetwork Dependent Convergence Function (SNDCF) may concatenate multiple packets, but this is presented as a single user data packet to the DLS. | |
| NOTE 2: UDATA DLPDUs consist of UINFO DLPDUs for broadcast of user data packets, and UCTRL DLPDUs for broadcast of LME data. UDATA is the broadcast equivalent of DATA and embraces all broadcast-type DLPDUs. | |

5.1.1.3 DATA DLPDU duplicate suppression and sequencing

| Requirement reference | |
|--|---|
| 5.1.1.3.1 | On a point-to-point connection, the receiving DLS sub-layer shall ensure that duplicated DATA DLPDUs are discarded and that all DATA DLPDUs which are part of a fragmented packet are delivered in the same order in which they appear in the packet. |
| NOTE: To facilitate duplicate suppression, a Toggle bit is included in the DLS DLPDU format. | |

5.1.1.4 Error detection

| Requirement reference | |
|--|---|
| 5.1.1.4.1 | The DLS shall rely on the MAC layer to ensure that DLPDUs corrupted during transmission are detected and discarded. |
| NOTE: A 16-bit CRC is provided in the burst format to support this error detection service. The MAC layer will reject corrupted packets. | |

5.1.1.5 Station identification

| Requirement reference | |
|--|---|
| 5.1.1.5.1 | A receiving station shall accept unicast DLPDUs addressed to its current station address. |
| NOTE: Unique source and destination addresses are included in the VDL Mode 4 DLS burst format in order to facilitate station identification. DLPDUs addressed to the current station address are routed to the DLS by the VSS. However, non-unique addressing is possible - with the resultant communications risk minimized through the assurance that any link address is locally unique. The ATN requires a unique address, hence non-unique addressing is not used with the ATN. | |

5.1.1.6 Broadcast addressing

| Requirement reference | |
|-----------------------|--|
| 5.1.1.6.1 | A VDL Mode 4 station shall accept broadcast DLPDUs and accept multicast DLPDUs that have been multicast to addresses to which it is listening. |

5.1.1.7 DLS Priority

| Requirement reference | |
|-----------------------|---|
| 5.1.1.7.1 | The DLS shall accept an indication of priority of the DATA DLPDUs as defined in table 5.8 of ETSI EN 302 842-2 [6]. |
| NOTE: | The DLS service user's selection of priority affects the QoS parameters used in the transfer of the DLS user packet as well as the queuing of the packet. |

5.1.1.8 DLS Link control DLPDUs

| Requirement reference | |
|-----------------------|--|
| 5.1.1.8.1 | For the purposes of link control, the DLS shall provide the following DLS DLPDU types: <ol style="list-style-type: none"> 1) ACK DLPDUs, consisting of INFO_ACK and CTRL_ACK, for the purposes of acknowledgement of DATA DLPDUs and DLS link control DLPDUs respectively. 2) RTS DLPDUs, consisting of CTRL_RTS, INFO_RTS and UDATA_RTS, for the purposes of making reservations for the transfer of DATA DLPDUs. 3) CTS DLPDUs, consisting of CTRL_CTS, INFO_CTS and UDATA_CTS, for purposes of acknowledging RTS DLPDUs and providing slots for subsequent transmission of DATA DLPDUs. 4) Other DLS link control DLPDUs, consisting of SZOM FRMR, FRMR_ACK, DM/DISC and DM/FRMR, for purposes of link initialization, reset and maintenance. |

5.1.2 DLS protocol specification

5.1.2.1 State variables

| Requirement reference | |
|-----------------------|--|
| 5.1.2.1.1 | The DLS shall maintain the state variables defined in table 5.1 for each data link between two peer DLEs. |
| NOTE: | If the link is reset for any reason, the DLS will discard any fragments associated with a partially-sent packet. |

Table 5.1: DLS state variables

| State variable | Usage |
|----------------|---|
| T_t | Current value of T bit (0 or 1) for transmitted DLPDUs. |
| T_r | Value of T bit (0 or 1) for last received DLPDU. |
| send array | An array storing user data packets and M-bit linked fragments queued for transmission (one per priority level). |
| receive array | An array storing received M-bit linked fragments queued for concatenation (one per priority level). |

5.1.2.2 DLS burst formats

| Requirement reference | |
|-----------------------|---|
| | DLS burst |
| 5.1.2.2.1 | A DLS station that implements the DLS protocol for communication with a peer station shall transmit the DLS burst defined in table 5.2 with the VSS user-supplied QoS and reservation parameters. |
| NOTE: | The DLS DLPDU field may continue past octet 10. |

Table 5.2: Normal unicast DLS burst format

| Description | Octet | Bit number | | | | | | | |
|---|-------|--|-----|-----|-----|---|---|---|---|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| message id, | 5 | res | res | res | res | 1 | 1 | 0 | 1 |
| DLS DLPDU | 6 | As per clauses 5.1.2.2.4 to 5.1.2.2.20 | | | | | | | |
| | 7 | | | | | | | | |
| | 8 | | | | | | | | |
| | 9 | | | | | | | | |
| | 10 | | | | | | | | |
| NOTE: All bits labelled "res" are reserved and shall be set to "0". | | | | | | | | | |

| Requirement reference | |
|---|--|
| 5.1.2.2.2 | The DLS burst shall consist of one or two DLS DLPDUs combined according to the procedures of clause 5.1.4.10. |
| 5.1.2.2.3 | A DATA DLPDU shall be the final field in the burst (and thus the burst can contain only one of these fields). |
| DLS DLPDU encoding | |
| 5.1.2.2.4 | The DLS DLPDU field shall indicate the DLPDU type and contain, as appropriate, the priority subfield, the More bit, the Toggle bit, the command/response status bit, the Initialize bit and the length subfield. |
| 5.1.2.2.5 | DATA DLPDUs shall consist of a single octet containing link control information and a variable length information field. |
| 5.1.2.2.6 | DATA DLPDUs shall be encoded as defined in table 5.3. |
| NOTE: The DLS burst will be able to combine up to two DLPDUs. DATA shall come last because it is a variable length DLPDU. | |

Table 5.3: Data DLPDU encoding

| Octet | n | | | | | | | | n+1 | | n+m |
|--|---|---|-----------------|-----------------|-----------------|-----------------|---|---|--------------------------------------|-------|-----|
| Bit | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | | | |
| CTRL | M | T | re | c/r | res | 0 | 0 | 0 | information field of length m octets | | |
| INFO | M | T | pr ₄ | pr ₃ | pr ₂ | pr ₁ | 1 | 0 | information field of length m octets | | |
| Reserved | X | X | X | X | 0 | 1 | 0 | 0 | information field of length m octets | | |
| Reserved | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | information field of length m octets | | |
| Reserved | X | X | X | 1 | 1 | 1 | 0 | 0 | information field of length m octets | | |
| Reserved | X | X | 1 | 0 | 1 | 1 | 0 | 0 | information field of length m octets | | |
| Reserved | X | 1 | 0 | 0 | 1 | 1 | 0 | 0 | information field of length m octets | | |
| Reserved | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | information field of length m octets | | |
| NOTE 1: "X" means 0 or 1; "M" is the More bit; "T" is the Toggle bit; "c/r" is the command/response status bit; "re" is the response expected bit; "pr" refers to priority; "n" is the DATA DLPDU octet. | | | | | | | | | | | |
| NOTE 2: In the case of the CTRL DLPDU the length (m) of the information field includes an additional two octets for the parameter ID and the parameter length (see octets n+1 and n+2 in table 5.13). | | | | | | | | | | | |

| Requirement reference | |
|-----------------------|---|
| 5.1.2.2.7 | RTS DLPDUs shall consist of two octets containing link control information. |
| 5.1.2.2.8 | RTS DLPDUs shall be encoded as defined in table 5.4. |

Table 5.4: Two-Octet DLPDUs encoding

| Octet | n | | | | | | | | n+1 | | | | | | | |
|-----------|------------------|------------------|------------------|------------------|---|---|---|---|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| Bit | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| CTRL_RTS | 0 | T | IB | 0 | 1 | 0 | 0 | 1 | res | res | res | res | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| INFO_RTS | 0 | T | 0 | 1 | 1 | 0 | 0 | 1 | pr ₄ | pr ₃ | pr ₂ | pr ₁ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| UDATA_RTS | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | pr ₄ | pr ₃ | pr ₂ | pr ₁ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| SZOM | neg ₄ | neg ₃ | neg ₂ | neg ₁ | 1 | 1 | 0 | 1 | seq ₈ | seq ₇ | seq ₆ | seq ₅ | seq ₄ | seq ₃ | seq ₂ | seq ₁ |
| Reserved | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | X | X | X | X | X | X | X | X |
| Reserved | 1 | X | X | 0 | 1 | 0 | 0 | 1 | X | X | X | X | X | X | X | X |
| Reserved | 1 | X | 0 | 1 | 1 | 0 | 0 | 1 | X | X | X | X | X | X | X | X |
| Reserved | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | X | X | X | X | X | X | X | X |
| Reserved | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | X | X | X | X | X | X | X | X |

NOTE: "X" means 0 or 1; "lg" refers to the length of the DATA burst to be sent expressed in slots; "T" is the Toggle bit; "IB" is the Initialize bit; "pr" refers to priority; "res" refers to bits available for the information field.

| Requirement reference | |
|-----------------------|--|
| 5.1.2.2.9 | The ACK, CTS and other DLS link control DLPDUs shall consist of one octet containing link control information. |
| 5.1.2.2.10 | These DLPDUs shall be encoded as defined in table 5.5. |

Table 5.5: Single octet DLPDUs encoding

| Octet | n | | | | | | | |
|-----------|---|-----|-----|---|---|---|---|---|
| Bit | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| UDATA_CTS | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| INFO_ACK | 0 | T | 0 | 1 | 0 | 0 | 0 | 1 |
| INFO_CTS | 0 | res | 0 | 1 | 1 | 0 | 1 | 1 |
| CTRL_ACK | 0 | T | res | 0 | 0 | 0 | 0 | 1 |
| CTRL_CTS | 0 | res | res | 0 | 1 | 0 | 1 | 1 |
| Reserved | 0 | X | X | 1 | 0 | 1 | 0 | 1 |
| FRMR_ACK | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| FRMR | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| DM/DISC | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| DM/FRMR | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| Reserved | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| Reserved | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| Reserved | 1 | X | 0 | 1 | 1 | 0 | 1 | 1 |
| Reserved | 1 | X | X | 0 | 0 | 0 | 0 | 1 |
| Reserved | X | X | X | 0 | 0 | 1 | 0 | 1 |
| Reserved | 1 | X | X | 0 | 1 | 0 | 1 | 1 |
| Reserved | X | X | 1 | 1 | 0 | 0 | 0 | 1 |
| Reserved | X | X | X | X | 0 | 0 | 1 | 1 |
| Reserved | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| Reserved | X | X | X | X | 0 | 1 | 1 | 1 |
| Reserved | X | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| Reserved | X | X | X | X | 1 | 1 | 1 | 1 |

NOTE 1: "X" means 0 or 1; T refers to the Toggle bit; FRMR to frame reject; DM/DISC to disconnect mode.

NOTE 2: All header bits labelled "res" are reserved and shall be set to "0".

| Requirement reference | |
|--|--|
| 5.1.2.2.11 | All reserved header bits (labelled "res") shall be set to zero on transmit and ignored on receipt. |
| 5.1.2.2.12 | A station receiving a reserved DLPDU from a peer with which it has a link shall reset the link in accordance with the procedures of clause 5.1.4.9. |
| 5.1.2.2.13 | A station receiving a reserved DLPDU from a peer with which it does not have a link shall either respond with a DM/DISC, DM/FRMR or simply ignore the DLPDU. |
| | Toggle bit |
| 5.1.2.2.14 | The T (Toggle) bit shall be alternately set to zero and one on each successful transmission (see note 1). |
| 5.1.2.2.15 | At the start of a communication between two stations, or when the link is reset, the toggle bit shall be initiated according to the procedures of clauses 5.1.4.2.15 to 5.1.4.2.29 for NSCOP communication. |
| | More bit |
| 5.1.2.2.16 | The M (More) bit shall be set to zero to indicate the end of a user data packet and to one to indicate that this fragment is not the last fragment in a multi-fragment user data packet and that further fragments will be transmitted (see note 2). |
| | Priority subfield |
| 5.1.2.2.17 | The priority subfield (pr) shall indicate the priority level of the transmission as defined in clause 5.1.1.6. |
| | Length subfield |
| 5.1.2.2.18 | The length subfield (lg) shall indicate the length of the DLS burst containing a DATA DLPDU in slots (see note 3). |
| 5.1.2.2.19 | It shall be encoded as one less than the absolute length. |
| | Initialize bit |
| 5.1.2.2.20 | Prior to sending a CTRL_RTS or upon receipt of a CTRL_RTS with IB (Initialize) Bit set to one the station shall initialize the T_t and T_r state variables and clear the send and receive arrays (see note 4). |
| | Negotiation subfield |
| 5.1.2.2.20a | The negotiation (neg) subfield shall indicate the link management parameters to be used for ZOCOP link control as defined in table 5.5a. |
| | SZOM Sequence subfield |
| 5.1.2.2.20b | The SZOM sequence (seq) subfield shall indicate the SZOM sequence number. |
| NOTE 1: The Toggle bit (T) is sufficient to provide duplicate detection and rejection. | |
| NOTE 2: The More bit (M) is set to zero if a user data packet is sent as a single fragment or on the last fragment of a fragmented; otherwise, it is set to one. The receiver reassembles a fragmented user data packet on reception before passing it to the user. | |
| NOTE 3: In the calculation of length, the size of the reservation protocol (default is response) and the effects of bit stuffing should be taken into account. A length of one ("1") slot would be encoded as 0000 binary and the maximum length of sixteen slots would be encoded as 1111 binary. | |
| NOTE 4: See clause 5.1.4.2 for the handling of INFO and CTRL DLPDUs in the process of being sent. | |

Table 5.5a: Interpretation of negotiation subfield

| Negotiation subfield | Parameters to use for ZOCOP link |
|----------------------|--|
| 0 | VDL Mode 4 default parameters for version 0. |
| 1 to 15 | Reserved for future use. |

| Requirement reference | |
|-----------------------|--|
| | Compressed combined RTS/INFO DLPDU encoding (type 1) |
| 5.1.2.2.21 | A DLS station wishing to send a combined RTS and INFO DLPDU according to the procedures of clause 5.1.4.10 when the priority of the RTS is different to that of the INFO packet shall transmit the compressed combined RTS/INFO (type 1) burst defined in table 5.6 with the VSS user supplied QoS and reservation parameters. |
| 5.1.2.2.22 | The T bit for the RTS shall be the inverse of the INFO bit. |

Table 5.6: Compressed combined RTS/INFO (type 1) burst format

| Description | Octet | Bit number | | | | | | | |
|---|-------|-------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| Message ID | 5 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| RTS priority and length | 6 | pr ₄ | pr ₃ | pr ₂ | pr ₁ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| INFO priority, M and T | 7 | M | T | pr ₄ | pr ₃ | pr ₂ | pr ₁ | res | res |
| Compressed RTS/INFO (type 1) DLPDU | 8 | Information field | | | | | | | |
| | 9 | | | | | | | | |
| | 10 | | | | | | | | |
| | 11 | | | | | | | | |
| | 12 | | | | | | | | |
| NOTE 1: The compressed combined RTS/INFO (type 1) DLPDU field may continue past octet 12. | | | | | | | | | |
| NOTE 2: 'res' refers to bits available for the information field. | | | | | | | | | |

| Requirement reference | |
|-----------------------|---|
| | Compressed combined RTS/INFO DLPDU encoding (type 2) |
| 5.1.2.2.23 | A DLS station wishing to send a combined RTS and INFO DLPDU according to the procedures of clause 5.1.4.10, when the priority of the RTS is the same as that of the INFO packet, shall transmit the compressed combined RTS/INFO (type 2) burst defined in table 5.7 with the VSS user supplied QoS and reservation parameters. |
| 5.1.2.2.24 | The T bit for the RTS shall be the inverse of the INFO bit and the priority the same as the INFO priority. |

Table 5.7: Compressed combined RTS/INFO (type 2) burst format

| Description | Octet | Bit number | | | | | | | |
|--|-------|-------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| Message ID | 5 | M | T | 1 | 0 | 0 | 1 | 0 | 1 |
| RTS/INFO priority and RTS length | 6 | pr ₄ | pr ₃ | pr ₂ | pr ₁ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| Compressed RTS/INFO (type 2) DLPDU | 7 | Information field | | | | | | | |
| | 8 | | | | | | | | |
| | 9 | | | | | | | | |
| | 10 | | | | | | | | |
| | 11 | | | | | | | | |
| NOTE 1: The compressed combined RTS/INFO (type 1) DLPDU field may continue past octet 11. | | | | | | | | | |
| NOTE 2: This burst format is intended to be used to link M-bit sequences where each fragment is part of the same user data packet and hence has the same priority. | | | | | | | | | |

5.1.3 DLS system parameters

| Requirement reference | |
|-----------------------|---|
| 5.1.3.1 | In addition to those defined in ETSI EN 302 842-2 [6], clause 5.3.2, the parameters needed by the DLS sublayer shall be as listed in table 5.8. |
| 5.1.3.2 | DLS parameters for NSCOP communications shall be determined during the exchange of CTRL DLPDUs, if the default values are not to be used. |
| 5.1.3.2a | DLS parameters for ZOCOP communications shall be determined by the exchange of the negotiation (neg) subfield within the SZOM DLPDU. |

Table 5.8: Data link service system parameters

| Symbol | Parameter name | Minimum | Maximum | Default | Increment |
|--|--|------------|--------------|--------------|-----------|
| TD1 | ZOCOP link transmit reset timer | 5 s | 600 s | 60 s | 1 s |
| TD2 | ZOCOP link receive reset timer | 10 s | 1 200 s | 90 s | 1 s |
| ND1 | Maximum number of octets in any user data packet | 143 octets | 2 063 octets | 1 511 octets | 1 octet |
| ND2 | Maximum length of short DLS transmission | 2 octets | 496 octets | 86 octets | 1 octet |
| ND3 | Maximum length of fragment | 1 slot | 16 slots | 5 slots | 1 slot |
| NOTE 1: The value of ND3 should be chosen such that the length of each DLS transmission containing the fragment is less than the maximum length of the DLS transmission defined by ND1 and greater than the maximum length of a short DLS transmission defined by ND2. | | | | | |
| NOTE 2: Parameter ND4 is defined in ETSI EN 302 842-2 [6]. | | | | | |

| Requirement reference | |
|---|---|
| Parameter TD1 (ZOCOP transmit reset timer) | |
| 5.1.3.2b | For a mobile station maintaining a ZOCOP link with another mobile station, Timer TD1 shall be reset when a DLPDU is sent to the peer. |
| 5.1.3.2c | Timer TD1 shall not be cancelled. |
| 5.1.3.2d | If Timer TD1 expires, then T_t shall be set to 0 and the send channel array shall be cleared. |
| Parameter TD2 (ZOCOP receive reset timer) | |
| 5.1.3.2e | For a mobile station maintaining a ZOCOP link with another mobile station, Timer TD2 shall be reset when a DLPDU is received from the peer. |
| 5.1.3.2f | Timer TD2 shall not be cancelled. |
| 5.1.3.2g | If Timer TD2 expires, the link shall be considered to be terminated. |
| NOTE: See clause 5.1.4.12 for an example of when the response reservation type is not used. | |

| Requirement reference | |
|---|--|
| Parameter ND1 (maximum number of octets of any user data packet) | |
| 5.1.3.3 | The parameter ND1 shall define the maximum number of octets in any user data packet that a DLS may accept from the data link user or from a peer station. |
| 5.1.3.4 | A station receiving a user data packet from a peer station greater in length than ND1 shall discard the packet and reset the link in accordance with the procedures of clause 5.1.4.9. |
| 5.1.3.5 | A station receiving a user data packet from a data link user greater in length than ND1 shall discard the packet (see note 1). |
| Parameter ND2 (maximum length of a short DLS transmission) | |
| 5.1.3.6 | The parameter ND2 shall define the maximum size in octets of a short DLS transmission including flags and reservation data that shall be sent using the short transmission procedures defined in clause 5.1.4.4. |
| 5.1.3.7 | A burst occupying x slots shall contain up to $23 + ((x-1) \times 63) / 2$ octets of data including reservation data, CRC and flags (see note 2). |
| Parameter ND3 (maximum length of fragment) | |
| 5.1.3.8 | The parameter ND3 shall define the maximum size in slots of a DLS burst. |
| NOTE 1: The maximum size of a user data packet for broadcast is set by parameter ND4. | |
| NOTE 2: A slot can contain 32 octets of data but the last slot in a sequence should only contain 24 octets to allow for propagation guard time. Allowing an average of one octet for every two slots for bit stuffing, one slot could contain 23 octets of data (including flags and reservation blocks); two slots could contain 54 octets; three slots could contain 86 octets; and so forth. | |

5.1.4 DLS procedures

5.1.4.1 Setting of re-transmission parameter

| Requirement reference | |
|---|--|
| 5.1.4.1.1 | For all DLS bursts containing CTRL, INFO, CTRL_RTS, INFO_RTS and UCTRL_RTS DLPDUs, the quality of service parameters Q5min, Q5max, Q5mult, Q5exp and Q5num shall be set as defined in table 5.9. |
| NOTE: Re-transmission of DLS DLPDUs may be handled in the VSS or DLS. | |

Table 5.9: Re-transmission parameters

| Symbol | Parameter name | | Default |
|--------|-------------------------------|----------------------------------|---------|
| Q5min | VSS retransmission parameters | Minimum | 1 s |
| Q5max | | Maximum | 15 s |
| Q5mult | | Multiplier | 1,45 |
| Q5exp | | Exponent | 1,7 |
| Q5num | | number of attempts | 4 |
| Q5wait | | maximum time to wait for a reply | 20 s |

5.1.4.2 Selection of user data packet for transmission

| Requirement reference | |
|--|---|
| User data packet priority | |
| 5.1.4.2.1 | A sending station shall maintain a prioritized queue of user data packets for transmission. |
| 5.1.4.2.2 | When determining which user data packet to transmit, the highest priority user data packet shall be sent first. |
| 5.1.4.2.3 | The DLS DLPDUs CTRL, and CTRL_RTS shall be classified as network management messages and given the highest priority. |
| 5.1.4.2.4 | The DLS DLPDUs INFO_RTS, and UDATA_RTS shall be assigned the same priority as the DATA DLPDU with which they are associated (see note). |
| User data packet fragmentation | |
| Determination of single or multiple fragment transmission | |
| 5.1.4.2.5 | If the length of the DLS burst containing a CTRL or INFO DLPDU is greater than ND2 octets, the sending station shall fragment the user data packet into one or more fragments of maximum size ND3, and format it according to the procedures of clauses 5.1.4.2.11 to 5.1.4.2.14. |
| 5.1.4.2.6 | Otherwise the user data packet shall be sent as a single fragment formatted according to the procedures of clauses 5.1.4.2.8 to 5.1.4.2.10. |
| 5.1.4.2.7 | The fragmentation of a user data packet shall take account of whether the station is combining a CTRL or INFO DLPDU with another DLS DLPDU in accordance with the procedures of clause 5.1.4.10. |
| Single fragment user data packet transmission | |
| 5.1.4.2.8 | A single fragment user data packet shall be transferred as a CTRL or INFO DLPDU. |
| 5.1.4.2.9 | The M bit shall be set to zero. |
| 5.1.4.2.10 | For an INFO DLPDU, the pr bits shall indicate the priority of the DLPDU. |
| Multiple fragment user data packet transmission | |
| 5.1.4.2.11 | A multiple fragment user data packet shall be transferred as a series of CTRL or INFO DLPDUs using the long transmission procedures defined in clause 5.1.4.5. |
| 5.1.4.2.12 | The M bit shall be set to 1 for all fragments except the last fragment. |
| 5.1.4.2.13 | The M bit shall be set to 0 for the last fragment. |
| 5.1.4.2.14 | For INFO DLPDUs, the pr bits shall indicate the priority of the DLPDU. |
| NOTE: All other DLS DLPDU types will be sent in pre-reserved slots (FRMR, DM/FRMR and DM/DISC) or will take the priority of the DATA packet with which they are combined (SZOM). Hence priority is not an issue. | |

| Requirement reference | |
|-----------------------|--|
| | Setting of T bit |
| | T bit initialization for NSCOP communication |
| 5.1.4.2.15 | When there is no established link (or link in the process of being established) between a mobile DLE and a ground DLE and the LME of either station requests the transmission of a CTRL user data packet, the sending DLE shall send the CTRL DLPDU using the long transmission procedures (see clause 5.1.4.5) (see notes 1 and 2). |
| 5.1.4.2.16 | In the CTRL_RTS, it shall set the IB bit to 1, the T bit to 0, and follow the procedures of clause 5.1.2.3.20. |
| 5.1.4.2.17 | On receipt of a CTRL_RTS DLPDU with IB = 1, the receiving DLE shall follow the procedures of clause 5.1.2.3.20. |
| 5.1.4.2.18 | The sender and receiver shall consider the link initialized. |
| 5.1.4.2.19 | The receiver shall immediately terminate any INFO transfers in progress. |
| 5.1.4.2.20 | Any partially received CTRL DLPDUs shall be discarded. |
| 5.1.4.2.21 | If any CTRL fragments had already been acknowledged, then the remainder of the CTRL DLPDU shall be abandoned. |
| 5.1.4.2.22 | However, if no CTRL_ACK had been received for a CTRL DLPDU, then its transfer shall continue unaffected. |
| 5.1.4.2.23 | The DLE shall consider the link connected upon direction from the LME (see note 3). |
| 5.1.4.2.24 | INFO, INFO_RTS, INFO_ACK and INFO_CTS DLPDUs shall only be sent on links that are connected. |
| 5.1.4.2.25 | Although a DLE may receive INFO DLPDUs (and generate INFO_ACKs), it shall not transmit INFO DLPDUs until it receives a CTRL_ACK to its CTRL (M=0). |
| 5.1.4.2.26 | On receipt of a CTRL_RTS, in a DLS burst addressed to it for which IB is equal to 1 and for which the T bit is equal to 1 a station shall send a DM/FRMR. |
| 5.1.4.2.27 | If a DLE with uninitialized state variables receives a CTRL_RTS DLPDU with IB equal to 0, then it shall respond with a DM/FRMR. |
| 5.1.4.2.28 | If a DLE receives an SZOM from a peer DLE, it shall respond with a DM/FRMR. |
| NOTE 1: | The DLE to which the mobile sent the CTRL_CMD may not be the DLE which responds with the CTRL_RSP. |
| NOTE 2: | INFO DLPDUs and partially sent or received CTRL DLPDUs are abandoned mid-transfer on the presumption that the peer station has restarted (there is no other legitimate reason for the IB = 1). |
| NOTE 3: | If the receiving LME indicates to the receiving DLE that the link is established after the receiving DLE's transmission of the CTRL_ACK, then the receiving DLE will not respond with a DM/FRMR to the INFO_RTS or INFO that it receives. |

| Requirement reference | |
|-----------------------|---|
| | T bit initialization for ZOCOP communication |
| 5.1.4.2.28a | A mobile sending station (the "sending station") wishing to send data to a mobile station (the "receiving station") for which it does not currently have a link, shall send a DLS burst containing a Start Zero Overhead Mode (SZOM) DLPDU and the first INFO (short transmission procedures) or INFO_RTS (long transmission procedures) DLPDU to the receiving station using respectively the short transmission procedures (see clause 5.1.4.4) or the long transmission procedures (see clause 5.1.4.5). |
| 5.1.4.2.28b | It shall set the negotiation subfield to the highest value that is supported by the DLS in the SZOM DLPDU, the sequence subfield to a value that it has not transmitted to the receiving station within the previous TD2 seconds, the T bit to 0 in the INFO DLPDU, and initialize its sending and receiving state variables. |
| 5.1.4.2.28c | On receipt of an unsolicited SZOM DLPDU with a sequence subfield different from the most recently received SZOM from the sending station, the receiving station shall initialize its sending and receiving state variables and consider the link established for receipt of further DLPDUs from the sending station. |
| 5.1.4.2.28d | The receiving station shall send a SZOM DLPDU combined with an INFO_ACK (short transmission procedures) or an INFO_CTS (long transmission procedures) in the slot reserved for the transmission setting T=0. |
| 5.1.4.2.28e | In the SZOM DLPDU, the receiving station shall set the negotiation field to the highest value that is supported by the DLS and the sequence subfield to 0. |
| 5.1.4.2.28f | The receiving station shall not transmit any INFO or INFO_RTS DLPDUs to the sending station prior to sending the SZOM/INFO_ACK or SZOM/INFO_CTS reserved transmission. |

| Requirement reference | |
|-----------------------|--|
| 5.1.4.2.28g | For both sending and receiving stations, link parameters shall be selected which correspond to the lower value of the negotiation subfields contained in the SZOM DLPDU sent by the sending station and in the SZOM DLPDU sent by the receiving station. |
| 5.1.4.2.28h | A station that transmitted an SZOM to a peer shall retransmit its initial transmission in response to any DLPDU other than an SZOM, DM/DISC, DM/FRMR, or general failure until it receives an SZOM. |
| 5.1.4.2.28i | After Q5num attempts, it shall report a failure to the DLE user. |
| | Transfer after initialization |
| 5.1.4.2.29 | When the T bit has been initialized, the sending station shall set the T bit for transmitted DLPDUs to the value of T_t . |

5.1.4.3 Selection of transmission procedures

| Requirement reference | |
|-----------------------|--|
| 5.1.4.3.1 | After a packet has been selected for transmission according to the procedures of clause 5.1.4.2 the sending station shall calculate the total length in octets of the DLS burst required to contain the DLPDU queued for transmission using the short transmission procedures defined in clause 5.1.4.4. |
| 5.1.4.3.2 | The total length shall include the length of any reservation fields contained within the DLS burst, together with any flags. |
| 5.1.4.3.3 | The calculation of the length of the DLS burst shall take account of whether the station will combine a DATA DLPDU with another DLS DLPDU in accordance with the procedures of clause 5.1.4.10. |
| 5.1.4.3.4 | If the total length is less than or equal to ND2, then the station shall use short transmission procedures (see clause 5.1.4.4) to transmit the queued data. |
| 5.1.4.3.5 | Otherwise, the station shall use the long transmission procedures (see clause 5.1.4.5). |
| 5.1.4.3.6 | The M bit shall be set to 0 for a short transmission. |
| | Recommendation |
| 5.1.4.3.7 | If there are other DLPDUs queued for transmission, then the station should also include an RTS per clause 5.1.4.10.4. |

5.1.4.4 Short transmission procedures

| Requirement reference | |
|--|--|
| | Transmission of DATA DLPDU |
| 5.1.4.4.1 | After the selection of a short transmission procedure, the sending station shall transmit a DLS burst containing the DATA DLPDU queued for transmission with QoS parameters as defined in table 5.10 using the random access procedures. |
| 5.1.4.4.2 | For CTRL and INFO DLPDUs, the T bit shall be set to T_t . |
| 5.1.4.4.3 | A burst containing a CTRL or INFO DLPDU shall contain a unicast request reservation field for the acknowledgement with the parameters in table 5.10. (See notes 1 and 2) |
| | Acknowledgement of DATA DLPDU |
| | Established link with sender |
| 5.1.4.4.4 | If a station receiving a CTRL or INFO DLPDU has an established link with the sender it shall transmit a DLS burst containing an ACK DLPDU in the slot reserved by the unicast request reservation field contained in the DATA DLPDU transmission (see note 3). |
| 5.1.4.4.5 | a) A CTRL_ACK DLPDU shall be sent in response to a CTRL DLPDU and an INFO_ACK DLPDU sent in response to an INFO DLPDU. |
| 5.1.4.4.6 | b) The T bit shall be set to the value of the T bit in the received DATA DLPDU. |
| NOTE 1: The priority field in the unicast reservation field is set equal to the priority setting in the INFO DLPDU as specified in table 5.10. | |
| NOTE 2: If the responder VSS sub-layer has not received a response from the VSS user in the scheduled reservation, it can send: | |
| a) a GENERAL FAILURE (see ETSI EN 302 842-2 [6] clause 5.2.19) with error type (err) 7E hex or FE hex; and | |
| b) a unicast request reservation according to the procedures of ETSI EN 302 842-2 [6] clause 5.2.20. | |
| NOTE 3: See clause 5.1.4.10 for an example of when the response reservation type is not used. | |

Table 5.10: Short transmission INFO DLPDU parameters

| Symbol | Parameter name | Default |
|--------|---|------------------------|
| V32 | Minimum response delay | (54 ms) x M1/60 slots |
| V33 | Maximum response delay | (5 s) x M1/60 slots |
| V34 | Source/destination control | 0 |
| V35 | Broadcast control | 0 |
| V36 | Length of reserved block | 1 slot |
| Q1 | Priority | Priority of INFO DLPDU |
| Q2a | Slot selection range constraint for level 1 | 150 NM |
| Q2b | Slot selection range constraint for level 2 | 150 NM |
| Q2c | Slot selection range constraint for level 3 | 0 NM |
| Q2d | Slot selection range constraint for level 4 | 300 NM |
| Q3 | Replace queued data | FALSE |
| Q4 | Number of available slots | 3 |

| Requirement reference | |
|-----------------------|---|
| | Non-receipt of acknowledgement |
| 5.1.4.4.7 | If an acknowledgement to a CTRL or INFO DLPDU is not received from the receiving station, the sending station shall retransmit the DLS burst containing the CTRL or INFO DLPDU and a unicast request reservation field according to the procedures of clause 5.3.2. |

5.1.4.5 Long transmission procedures

| Requirement reference | |
|--|--|
| | Transmission of request to send |
| 5.1.4.5.1 | After the selection of a long transmission procedure, the sending station shall transmit a DLS burst containing an RTS DLPDU to the receiving station in the transmit queue with QoS parameters as defined in table 5.11. |
| 5.1.4.5.2 | When using the long transmission procedure: <ol style="list-style-type: none"> A station shall send a CTRL_RTS, INFO_RTS or UDATA_RTS DLPDU if a CTRL, INFO or UDATA DLPDU respectively is to be transmitted. |
| 5.1.4.5.3 | <ol style="list-style-type: none"> The pr subfield shall indicate the priority of the INFO_RTS and UDATA_RTS DLPDUs to be transmitted (see clause 5.1.1.6). |
| 5.1.4.5.4 | <ol style="list-style-type: none"> The length subfield shall indicate the length of the DLS burst required to contain the DATA DLPDU (see clause 5.1.2.3.18). |
| 5.1.4.5.5 | <ol style="list-style-type: none"> For CTRL_RTS and INFO_RTS DLPDUs, the T bit shall be set to Tt. |
| 5.1.4.5.6 | Each burst shall contain a unicast request reservation field with the parameters set as defined in table 5.11. |
| NOTE 1: The ground may use the priority (pr) subfield to manage the link resource during congestion. | |
| NOTE 2: Transfer of broadcast data using the long transmission procedures involves directing the UDATA_RTS at a specific station, receiving a set of slots from that station via the UDATA_CTS, and then broadcasting the UDATA in the reserved slots. | |
| NOTE 3: The priority field in the unicast request reservation field is set equal to the priority setting in the INFO DLPDU as specified in table 5.11. | |

Table 5.11: Long transmission RTS DLPDU parameters

| Symbol | Parameter name | Default |
|--------|---|-----------------------|
| V32 | Minimum response delay | (54 ms) x M1/60 slots |
| V33 | Maximum response delay | (5 s) x M1/60 slots |
| V34 | Source/destination control | 0 |
| V35 | Broadcast control | 0 |
| V36 | Length of reserved block | 1 slot |
| Q1 | Priority | Priority of RTS |
| Q2a | Slot selection range constraint for level 1 | 150 NM |
| Q2b | Slot selection range constraint for level 2 | 150 NM |
| Q2c | Slot selection range constraint for level 3 | 0 NM |
| Q2d | Slot selection range constraint for level 4 | 300 NM |
| Q3 | Replace queued data | FALSE |
| Q4 | Number of available slots | 3 |

| Requirement reference | |
|-----------------------|--|
| | Response to RTS |
| | Response if a DATA DLPDU has not previously been received |
| 5.1.4.5.7 | If the responder has an established link with the sender and the received DLPDU is a UDATA_RTS or if the T bit within the CTRL_RTS or INFO_RTS DLPDU is not equal to T_r , then the receiving station shall transmit a CTS DLPDU in a DLS burst in the slot reserved by the RTS DLPDU. |
| 5.1.4.5.8 | A CTRL_CTS, INFO_CTS or UDATA_CTS DLPDU shall be sent in response to a CTRL_RTS, INFO_RTS or UDATA_RTS DLPDU respectively. |
| 5.1.4.5.9 | For CTRL_CTS and INFO_CTS DLPDUs, the burst shall contain an information transfer request reservation field and be transmitted with the parameters in table 5.12, indicating the number of slots reserved for transfer of the DATA DLPDU. |
| 5.1.4.5.10 | For a UDATA_CTS DLPDU, the burst shall contain a unicast request reservation field and be transmitted with the parameters in table 5.11, indicating the number of slots reserved for transfer of the DATA DLPDU. |
| | Response if a DATA DLPDU has previously been received |
| 5.1.4.5.11 | If the responder has an established link with the sender and if the T bit within the CTRL_RTS or INFO_RTS DLPDU is equal to T_r , then the receiving station shall transmit an ACK DLPDU in the slot reserved by the RTS. |
| 5.1.4.5.12 | a) A CTRL_ACK or INFO_ACK DLPDU shall be sent in response to a CTRL_RTS or INFO_RTS DLPDU respectively. |
| 5.1.4.5.13 | b) The T bit shall be set to the value of the T bit in the received DATA DLPDU. |
| 5.1.4.5.14 | The DLS burst containing the ACK DLPDU shall contain a response reservation type except as determined by the procedures of clause 5.1.4.10. |
| | Channel too busy |
| 5.1.4.5.15 | If the channel is too busy (either the receiving station cannot find a sufficiently large series of contiguous slots or the priority is too low for the channel utilization), then the receiving station shall transmit either: a general confirm including a unicast reservation with $V_{34} = 1$ indicating when the responder will transmit an information transfer request in response to the RTS; or, a general failure (clause 5.2.20 of ETSI EN 302 842-2 [6]) with error type = 01 hex. |
| NOTE: | If the responder sends a GENERAL FAILURE (see ETSI EN 302 842-2 [6] clause 5.2.20), the sender can re-transmit the RTS after the time out defined by the back-off delay or in the slot reserved by the destination. |

Table 5.12: CTS DLPDU parameters

| Symbol | Parameter name | Default |
|--------|---|---|
| V42 | Length of information transfer | Sufficient to include requested INFO DLPDUs |
| V43 | Minimum information transfer delay | (54 ms) \times M1/60 slots |
| V44 | Maximum information transfer delay | (5 s) \times M1/60 slots |
| V45 | Minimum response delay | (54 ms) \times M1/60 slots |
| V46 | Maximum response delay | (5 s) \times M1/ slots |
| Q1 | Priority | Priority of RTS |
| Q2a | Slot selection range constraint for level 1 | 150 NM |
| Q2b | Slot selection range constraint for level 2 | 150 NM |
| Q2c | Slot selection range constraint for level 3 | 0 NM |
| Q2d | Slot selection range constraint for level 4 | 300 NM |
| Q3 | Replace queued data | FALSE |
| Q4 | Number of available slots | 3 |

| Requirement reference | |
|---|--|
| | Response to CTS |
| | Transmission of DATA DLPDU |
| 5.1.4.5.16 | On receipt of a CTRL_CTS or INFO_CTS DLPDU in a DLS burst addressed to it and with an information transfer request reservation field a station shall transmit the requested DATA DLPDU in the allocated reservation with the T bit set to T_t . |
| 5.1.4.5.17 | The DLS burst containing the DATA DLPDU shall contain a response reservation type. |
| 5.1.4.5.18 | On receipt of a UDATA_CTS DLPDU in a DLS burst addressed to it with a unicast request reservation field a station shall transmit the requested UDATA DLPDU in the allocated reservation. |
| | Response if no information to transmit |
| 5.1.4.5.19 | If upon receipt of a CTS the station has nothing to transmit (e.g. after a reset), it shall transmit one of the following: <ul style="list-style-type: none"> a) an FRMR if the link is connected; b) a DM/DISC if the link is disconnected; c) a DM/FRMR if the link is in the process of connecting. |
| | Recommendation |
| 5.1.4.5.20 | If the station has not transmitted an INFO DLPDU and a higher priority user data packet arrived after the RTS had been transmitted, the station should transmit as much of the highest priority packets as will fit in the current reservation with the same value for T as contained in the RTS. |
| | Acknowledging the data |
| | DATA DLPDU received |
| 5.1.4.5.21 | A receiving station which transmitted a CTRL_CTS or INFO_CTS DLPDU in a DLS burst containing an information transfer request reservation field (and consequently has a reservation for an acknowledgement) and which has received the DATA DLPDU with a T bit not equal to T_r , shall follow the procedures of clauses 5.1.4.4.4 to 5.1.4.4.6. |
| | Response if DATA DLPDU not received |
| 5.1.4.5.22 | A receiving station which transmitted a CTRL_CTS or INFO_CTS DLPDU in a DLS burst containing an information transfer request reservation field (and consequently has a reservation for an acknowledgement) and which has not received the CTRL or INFO DLPDU with a T bit not equal to T_r , shall transmit in the slot reserved for an acknowledgement an ACK DLPDU in a DLS burst. |
| 5.1.4.5.23 | a) A CTRL_ACK or INFO_ACK DLPDU shall be sent in response to a CTRL or INFO DLPDU respectively. |
| 5.1.4.5.24 | b) The T bit shall be set to the inverse of the value of the T bit in the last received RTS DLPDU (see note). |
| 5.1.4.5.25 | The DLS burst containing the ACK DLPDU shall contain a response reservation type. |
| NOTE: The use of the Toggle bit (T) not equal to T_r indicates a negative acknowledgement (NACK). | |

5.1.4.6 No link with sender

| Requirement reference | |
|--|---|
| 5.1.4.6.1 | If the responder to any DLPDU other than a CTRL_RTS DLPDU with IB equal to 1 or an SZOM DLPDU (which may be combined with an INFO or INFO_RTS DLPDU) neither has nor is attempting to establish a link with the sender, the responder shall send a DLS burst containing a DM/DISC DLPDU in the slot reserved by the unicast or information transfer request reservation field contained in the data DLPDU transmission. |
| 5.1.4.6.2 | If the responder is trying to establish a link with the sender, then it shall respond with a DM/FRMR to any DLPDU other than a CTRL-related DLPDU or SZOM DLPDU for NSCOP and ZOCOP respectively. |
| NOTE: Per clause 5.1.4.2.15, an NSCOP link is considered established when a DLE sends or receives the last fragment of a CTRL_RSP. Consequently, a DLE may be retransmitting the last fragment of a CTRL_RSP whilst it is acknowledging INFO DLPDUs. | |

5.1.4.7 User data packet reception

| Requirement reference | |
|---|--|
| Receipt and forwarding of received DATA DLPDUs | |
| 5.1.4.7.1 | When a CTRL or INFO DLPDU is received without errors from another station, the value of the T bit shall be inspected and recorded. |
| 5.1.4.7.2 | If this is: a) an INFO DLPDU with the T bit set to zero combined with a SZOM DLPDU; or b) the value of the T bit is not equal to T_r then the user data packet or user data packet fragment shall be accepted and T_r set to the value of the T bit in the received INFO or CTRL DLPDU. |
| 5.1.4.7.3 | Otherwise, the received user data packet or user data packet fragment shall be discarded as a duplicate. |
| Concatenation of multiple fragment user data packets | |
| 5.1.4.7.4 | If any preceding user data packet fragments have been received with, in the case of an INFO DLPDU, the same value of pr subfield and with the M bit set to one, then the user data part of the received DATA DLPDU shall be concatenated to the end of the user data packet fragments. |
| 5.1.4.7.5 | If the M bit is set to zero on the received DATA DLPDU, then the user data part of the received DATA DLPDU, including any user data packet fragments received earlier and with which it has been concatenated, shall be passed to the service user as a single incoming user data packet. |
| Unacknowledged DLPDUs | |
| 5.1.4.7.6 | DM/DISC, DM/FRMR and all ACK DLPDUs shall be unacknowledged. |

5.1.4.8 Receipt of ACK DLPDU

| Requirement reference | |
|--|--|
| Receipt of an expected ACK DLPDU | |
| 5.1.4.8.1 | When an ACK DLPDU is received without errors from another station and there was an outstanding DATA DLPDU to be acknowledged, the value of the T bit shall be inspected and the following operations performed. |
| 5.1.4.8.2 | If T is equal to T_t , then the DATA DLPDU shall be assumed successfully received and the value of T_t set to the inverse of the current value of T_t . |
| 5.1.4.8.3 | If the ACK DLPDU is not received by the expected reserved slot, then the sending station shall re-send the DATA DLPDU using the short or long transmission procedures as determined by the procedures of clause 5.1.4.3. (see note 1). |
| 5.1.4.8.4 | When a DATA DLPDU has been successfully received, the highest priority fragment (either the next fragment of the current user data packet or the first fragment/complete DLPDU of the next user data packet) in the send queue, if any, shall be selected for transmission using the procedures of clause 5.1.4.2. (see note 2). |
| Receipt of an unexpected ACK DLPDU | |
| 5.1.4.8.5 | When an ACK DLPDU is received without errors from another station and there was no outstanding DATA DLPDUs to be acknowledged, the value of the T bit shall be inspected and the following operations performed. |
| 5.1.4.8.6 | If T is equal to T_t then the link shall be reset as per clause 5.1.4.9. |
| 5.1.4.8.7 | If T is not equal to T_t then the ACK shall be ignored. |
| NOTE 1: Retransmission is based on the expected receipt time and not on the receipt of a NACK. | |
| NOTE 2: The selection of highest priority allows the station to pre-empt a lower priority transfer (e.g. more bit (M) linked fragments) with a higher priority user data packet or set of fragments. | |

5.1.4.9 Link reset

| Requirement reference | |
|--|---|
| Link reset during link setup | |
| 5.1.4.9.1 | The sending station shall disconnect the link by sending a DM/FRMR DLPDU, which is transmitted in the reservation placed for the reply. |
| 5.1.4.9.2 | Upon receipt of a DM/FRMR DLPDU, the receiving station shall disconnect the link. |
| Link reset of an established link | |
| 5.1.4.9.3 | The sending station shall reset the link by sending a FRMR DLPDU in a DLS burst placing a unicast reservation for the response. |
| 5.1.4.9.4 | The sending station shall discard all outstanding user data packets in the send and receive arrays. |
| 5.1.4.9.5 | If a FRMR_ACK is not received in the reserved slot, the FRMR shall be sent again using the re-transmission procedures. |
| 5.1.4.9.6 | Only when a FRMR_ACK has been received shall the station attempt to re-send data to the receiving station using the procedures of clause 5.1.4.3. |
| 5.1.4.9.7 | If the sending station receives an INFO or RTS DLPDU from the peer whilst waiting for a FRMR_ACK, it shall send an FRMR in response. |
| 5.1.4.9.8 | Upon receipt of an unsolicited FRMR_ACK, a DLE shall respond with an FRMR. |
| 5.1.4.9.9 | On receipt of a FRMR DLPDU, the receiving station shall discard any outstanding fragments in the receive array and transmit a FRMR_ACK DLPDU in a DLS burst in the reserved slot. |

5.1.4.10 Linking DLS DLPDU transmissions

| Requirement reference | |
|--|---|
| 5.1.4.10.1 | An implementation of the DLS shall be capable of providing a combined DLPDU response even if the station does not initiate the use of combined DLPDUs. |
| | Recommendation |
| 5.1.4.10.2 | A station with a queue of transmissions to send to a receiving station should link transmissions using the procedures set out in this clause. |
| | Allowed DLPDU combinations |
| 5.1.4.10.3 | It shall be possible to combine the following DLPDUs (see note 1): <ul style="list-style-type: none"> - RTS/DATA; - ACK/CTS; - ACK/DATA; - ACK/RTS; - SZOM/other DLPDUs as described in clauses 5.1.4.2.28a to 5.1.4.2.28i. |
| | Recommendation: Combined RTS/DATA DLPDUs |
| 5.1.4.10.4 | When a receiving station has selected a user data packet for transmission using the procedures of clause 5.1.4.2 it should also select the next user data packet with the highest priority and place an RTS DLPDU in the DLS burst containing the DATA DLPDU for the first user data packet, setting the T bit in the RTS to the inverse of T_i and append a response reservation type (see notes 2 and 3). |
| | Combined ACK/CTS DLPDUs |
| 5.1.4.10.5 | On receipt of a RTS/DATA DLS burst, a station shall process the DATA DLPDU first according to the procedures of clause 5.1.4.5.21. |
| 5.1.4.10.6 | If the station can find sufficient resources for a subsequent DATA DLPDU transfer, then the DLS burst containing the ACK DLPDU shall also contain a CTS DLPDU for the next DATA DLPDU transfer. |
| 5.1.4.10.7 | A CTRL_CTS, INFO_CTS or UDATA_CTS DLPDU shall be sent in response to a CTRL_RTS, INFO_RTS or UDATA_RTS DLPDU respectively. |
| 5.1.4.10.8 | In the case of CTRL_CTS and INFO_CTS DLPDUs, instead of the response reservation type required by the procedures of clause 5.1.4.5.21 the burst shall contain an information transfer request reservation field transmitted using the parameters defined in table 5.12, indicating the number of slots reserved for transfer of the DATA DLPDU. |
| | Combined DATA/ACK DLPDUs |
| 5.1.4.10.9 | A receiving station which has data that would fit into a single slot DLS burst containing an ACK DLPDU to send back to the sending station, shall include its own DATA DLPDU in this DLS burst. |
| 5.1.4.10.10 | The burst shall contain a unicast request reservation field with the parameters in table 5.11 for the acknowledgement of the DATA DLPDU (see note 4). |
| | Combined ACK/RTS DLPDUs |
| 5.1.4.10.11 | If a receiving station has data to send back to the sending station which cannot fit into a single slot DLS burst containing an ACK DLPDU, then the station shall combine an RTS DLPDU for its own DATA DLPDU in the DLS burst containing the ACK DLPDU and use the long transmission procedures for the data transfer. |
| | Recommendation - Combined ACK/RTS DLPDUs to different peers |
| 5.1.4.10.12 | If a station is sending an ACK DLPDU to one destination and has data to send to a different destination, then the station should include a unicast reservation field with $sdf = 1$ with the DLS burst containing the ACK and then transmit an RTS DLPDU to the new destination in the reserved slot (see note 5). |
| NOTE 1: Receipt of a combined DLPDU not listed in this clause is handled in an implementation-defined manner. | |
| NOTE 2: Because the DATA DLPDU is unlimited in length, the RTS shall precede the DATA DLPDU. However, since the RTS contains the inverse T bit of the transmitted DATA DLPDU (as it is for the subsequent DATA DLPDU), it shall be processed second. | |
| NOTE 3: This recommendation also applies in cases where the short transmission procedures would normally be selected for transmission but for which there is an opportunity to combine it with the end of the previous long transmission procedure. | |
| NOTE 4: A station may combine a transport acknowledgement with the ACK if this can be produced in time. | |
| NOTE 5: This makes it possible for a ground station to link a series of transmissions to different destinations. | |

5.1.4.11 CTRL DLPDU

| Requirement reference | |
|-----------------------|--|
| 5.1.4.11.1 | The CTRL DLPDU shall be used for the LME to establish and maintain links as defined in clause 5.2. |

5.1.4.12 Procedures for air-air communication

| Requirement reference | |
|--|---|
| 5.1.4.12.1 | Air-air communication services shall be supported by a Zero-Overhead Connection Oriented Protocol (ZOCOP). |
| Air-air data exchange after link initialization | |
| 5.1.4.12.2 | When a ZOCOP link has been established, data shall be exchanged as determined by either the short or long transmission procedures as described in clauses 5.1.4.4 and 5.1.4.5. |
| Link maintenance | |
| 5.1.4.12.3 | A station that has established a ZOCOP link with a peer, and which has not sent a packet to that peer for TD1 seconds, and that wishes to send a DATA packet to the peer, shall re-establish the link according to the procedures of clauses 5.1.4.2.28a to 5.1.4.2.28i. |
| 5.1.4.12.4 | A station shall regard the link as terminated if it has not transmitted to the same peer for TD1 seconds and if it has not received a transmission from the same peer for TD2 seconds. |
| 5.1.4.12.5 | A station shall regard the link as terminated if it receives a DM/DISC DLPDU or a DM/FRMR from the peer station or an attempt to send a packet to the peer station fails because of a failure notified by the retransmission procedures of clause 5.3.4. |
| Action on receipt of a CTRL parameter in a ZOCOP link | |
| 5.1.4.12.6 | If a CTRL_RTS (IB = 0), CTRL_CTS, CTRL, or CTRL_ACK DLPDU is received on an established ZOCOP link, then the receiver shall respond with a DM/FRMR in the reserved slot. |
| 5.1.4.12.7 | If a CTRL_RTS (IB = 1) is received on an established ZOCOP link, then the old link shall be considered disconnected and the receiver shall either proceed to establish the link via NSCOP procedures or respond with a general failure with a cause code indicating policy reasons. |

5.2 Link management entity sublayer

5.2.1 Services

5.2.1.1 General

| Requirement reference | |
|-----------------------|--|
| 5.2.1.1.1 | The services of the LME shall be as follows: a) link provision; and b) link change notification. |

5.2.1.2 Link provision

| Requirement reference | |
|-----------------------|---|
| 5.2.1.2.1 | Each Ground System and each Mobile System supporting air/ground point-to-point communication services shall include the functionality of a VDL Management Entity (VME). |
| 5.2.1.2.2 | A VME shall be responsible for the data link management policy of the System. |
| 5.2.1.2.2a | In a Mobile System the VME shall be responsible for determining with which Ground System(s) the Mobile System maintains datalink communications, at any given time. |
| 5.2.1.2.3 | Void. |
| 5.2.1.2.4 | A VME shall have a Link Management Entity (LME) for each peer LME. |
| 5.2.1.2.5 | Void. |
| 5.2.1.2.5a | A mobile VME shall have an LME per ground system. |
| 5.2.1.2.6 | An LME shall establish a link between a local DLE and a remote DLE associated with its peer LME. |
| 5.2.1.2.7 | Void. |
| 5.2.1.2.7a | A mobile LME shall determine if a ground station is associated with its peer ground LME by bit-wise logical ANDing the DLS address with the station ground system mask provided by the peer ground LME; two ground stations with identical masked DLS addresses are associated with the same LME. |
| 5.2.1.2.8 | Each ground and mobile LME shall monitor all transmissions (both DLS and VSS) from its peer's stations to maintain a reliable link between some ground station and a mobile while the mobile is in coverage of an acceptable ground station in the ground system. |

5.2.1.3 Link change notifications

| Requirement reference | |
|---|---|
| 5.2.1.3.1 | The VME shall notify the intermediate system-system management entity (IS-SME) of changes in link connectivity supplying information contained in the CTRL DLPDUs received. |
| NOTE: The IS-SME is a constituent entity in the ATN router, whose functions include managing the route initiation and termination procedures by responding to changes in subnetwork connectivity. | |

5.2.1.4 CTRL DLPDU

| Requirement reference | |
|-----------------------|---|
| 5.2.1.4.1 | Link management data directed to a particular peer shall be contained within a DLS burst encoded within the information field of a CTRL DLPDU (see table 5.3) as defined in table 5.13. |

Table 5.13: CTRL DLPDU format

| Description | Octet | Bit number | | | | | | | |
|---|----------|----------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| CTRL DLPDU header | n | M | T | re | c/r | res | 0 | 0 | 0 |
| CTRL parameter 1: Parameter ID | n+1 | id ₈ | id ₇ | id ₆ | id ₅ | id ₄ | id ₃ | id ₂ | id ₁ |
| Parameter length | n+2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| Parameter value | n+3 | q1 ₈ | q1 ₇ | q1 ₆ | q1 ₅ | q1 ₄ | q1 ₃ | q1 ₂ | q1 ₁ |
| | to | to | | | | | | | |
| | n+2 + lg | qlg ₈ | qlg ₇ | qlg ₆ | qlg ₅ | qlg ₄ | qlg ₃ | qlg ₂ | qlg ₁ |
| | | more CTRL parameters | | | | | | | |
| NOTE: 'res' refers to bits available for the information field. | | | | | | | | | |

| Requirement reference | |
|-----------------------|---|
| 5.2.1.4.2 | The settings of the M, IB and T bits shall be as defined in clause 5.1. |
| 5.2.1.4.3 | The settings of the c/r and re bits shall be as defined in table 5.17. |

5.2.1.5 Broadcast link management burst

| Requirement reference | |
|---|---|
| 5.2.1.5.1 | ucid = 3 shall indicate that the UCTRL DLPDU contains a broadcast CTRL_CMD with the values of c/r and re, defined in clauses 5.2.2.2.6 to 5.2.2.2.10, assumed to be equal to 0. |
| 5.2.1.5.2 | ucid = 4 indicates that the UCTRL DLPDU contains a UCTRL_DM parameter only as defined in clauses 5.2.2.2.30 to 5.2.2.2.34. |
| NOTE: For ucid=4, the parameter ID and parameter length fields are absent from the UCTRL DLPDU as defined in clause 5.2.2.2.33. | |

5.2.2 Control (CTRL) parameter formats

5.2.2.1 Encoding

| Requirement reference | |
|--|---|
| 5.2.2.1.1 | The CTRL parameters described in this clause and also in ETSI EN 302 842-2 [6], clause 5.4.3 and ETSI EN 302 842-3 [7], clause 5.1.5 shall be included in the user data field of CTRL and UCTRL DLPDUs. |
| NOTE 1: The tables in this clause are divided into three major columns that define the field name, the bit encoding and brief explanatory notes. | |
| NOTE 2: The CTRL parameters are extracted from the DLS sub-layer CTRL DLPDUs and sent to the LME. | |

5.2.2.2 General purpose information parameters

| Requirement reference | |
|---|--|
| 5.2.2.2.1 | VDL Mode 4 shall use the CTRL parameters defined below to allow mobile and ground-based LMEs to transfer basic information to each other. |
| Parameter set identifier | |
| 5.2.2.2.2 | The parameter set identifier shall be sent first in a group of non-VDL Mode 4 parameters and indicates to which set the parameters belong. |
| 5.2.2.2.3 | The parameter set identifier shall be encoded as per table 5.14. |
| 5.2.2.2.4 | There shall be no VDL Mode 4 parameter set identifier. |
| 5.2.2.2.5 | VDL Mode 4 parameters shall always be sent before any other parameter groups (see note). |
| NOTE: Parameter ID 00hex means change to different set, in which case the parameter value contains the identity of the new parameter set. A parameter set identity starting with "X" (equals 58 hex) is proprietary and is not defined in ICAO Technical Manuals. Parameter ID FFhex is reserved for upper layer data, in which case the parameter value contains upper layer data. | |

Table 5.14: Parameter set identifier encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|--------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | id ₈ | id ₇ | id ₆ | id ₅ | id ₄ | id ₃ | id ₂ | id ₁ | Parameter set identifier |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | q ₈ | q ₇ | q ₆ | q ₅ | q ₄ | q ₃ | q ₂ | q ₁ | |

| Requirement reference | |
|--|--|
| Connection management parameter | |
| 5.2.2.2.6 | The connection management parameter shall define the type of CTRL sent and the connection options negotiated for that particular link. |
| 5.2.2.2.7 | The connection management parameter shall be used in CTRL DLPDUs sent during link establishment and ground-initiated ground station handoff. |
| 5.2.2.2.8 | The connection management parameter shall be encoded as per tables 5.15 to 5.17. |
| 5.2.2.2.9 | An LME shall set the reserved bits to 0 on transmission. |
| 5.2.2.2.10 | An LME shall ignore the value of the reserved bits on receipt. |

Table 5.15: Connection management parameter

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Connection management |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | 0 | 0 | 0 | 0 | 0 | 0 | r | h | |

NOTE: The value in the parameter length field is variable to allow for the possibility of additional options.

Table 5.16: Connection management parameter values

| Bit | Name | Encoding | |
|--------|----------|----------|--------------------------------|
| 1 | h | h = 0 | No link currently established. |
| | | h = 1 | Link currently established. |
| 2 | r | r = 0 | Link connection accepted. |
| | | r = 1 | Link connection refused. |
| 3 to 8 | Reserved | Set to 0 | |

Table 5.17: Abbreviated CTRL names

| Name | c/r | re | h | r | Notes |
|--------------|-----|----|---|---|--|
| GSIF | - | - | - | - | Ground Station Identification Frame - sent using format defined in clause 5.2.1.5. |
| CTRL_CMD_LE | 0 | 1 | 0 | 0 | |
| CTRL_CMD_LCR | 0 | 0 | 0 | 1 | Link Connection Refused. |
| CTRL_CMD_HO | 0 | 1 | 1 | 0 | If re=1, then Initiating Handoff. |
| CTRL_CMD_HO | 0 | 0 | 1 | 0 | If broadcast and re=0, then commanding a Broadcast Handoff. If unicast and re=0, then Requesting Handoff. |
| CTRL_RSP_LE | 1 | 1 | 0 | 0 | |
| CTRL_RSP_LCR | 1 | 1 | 0 | 1 | |
| CTRL_RSP_HO | 1 | 1 | 1 | 0 | |

| Requirement reference | |
|----------------------------------|--|
| CTRL sequencing parameter | |
| 5.2.2.2.11 | The CTRL sequencing parameter shall define the CTRL sequence number (sss) and a CTRL retransmission number (rrr). |
| 5.2.2.2.12 | The CTRL sequencing parameter shall be encoded as per table 5.18. |
| 5.2.2.2.13 | An LME shall increment the sequence number for every new CTRL (setting the retransmission field to 0 on the first transmission). |
| 5.2.2.2.14 | An LME shall increment the retransmission field after every retransmission. |
| 5.2.2.2.15 | In a CTRL_RSP, the sequence number shall be set to the value of the CTRL_CMD sequence number generating the response. |
| 5.2.2.2.16 | In a CTRL_RSP the retransmission field shall be set to 0 when transmitting and ignored when received. |

Table 5.18: CTRL sequencing parameter

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | CTRL sequencing |
| Parameter length | n + 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | |
| Parameter value | n + 3 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ | |

NOTE: 'r' refers to re-transmission number; 's' refers to sequence number.

| Requirement reference | |
|-----------------------|---|
| | Protocol options parameter |
| 5.2.2.2.17 | The protocol options parameter shall define which protocol options are supported by the transmitting station. |
| 5.2.2.2.18 | The protocol options parameter shall be encoded as per tables 5.19 and 5.20. |
| 5.2.2.2.19 | When both this parameter and the Connection Management parameter are included in a CTRL, the bit values for those options which are included in both parameters shall be determined by the Connection Management parameter. |

Table 5.19: Protocol options parameter

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|------------|---|---|---|---|---|---|---|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | Protocol options |
| Parameter length | n + 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | |
| Parameter value | n + 3 | 0 | 0 | 0 | 0 | 0 | b | i | 0 | |

NOTE: 'b' refers to broadcast link handoff, 'i' refers to initiated handoff.

Table 5.20: Protocol options parameter values

| Bit | Name | Encoding | |
|-----------|----------|----------|---------------------------------------|
| 2 | i | i = 0 | Does not support initiated Handoff. |
| | | i = 1 | Supports initiated Handoff. |
| 3 | b | b = 0 | Broadcast link handoff not supported. |
| | | b = 1 | Broadcast link handoff supported. |
| 1, 4 to 8 | Reserved | Set to 0 | |

| Requirement reference | |
|-----------------------|---|
| | LCR cause parameter |
| 5.2.2.2.20 | The LCR cause parameter shall define the reason why the link connection request was refused. |
| 5.2.2.2.21 | The parameter, which may be repeated, shall consist of a rejection cause code (c bits), backoff delay time in seconds (d bits), and any additional data required by the various parameters. |
| 5.2.2.2.22 | The LCR cause parameter shall be encoded as per table 5.21. |
| 5.2.2.2.23 | Cause codes 00 hex to 7F hex shall apply to the responding station. |
| 5.2.2.2.24 | Cause codes 80 hex to FF hex shall apply to the responding system. |
| 5.2.2.2.25 | Cause codes shall be encoded as per table 5.22. |
| 5.2.2.2.26 | At least one copy of this parameter shall be included whenever the "r" bit in the Connection Management parameter is set to 1. |
| 5.2.2.2.27 | This parameter shall not be included if the "r" bit is set to 0. |
| 5.2.2.2.28 | An LME receiving an LCR Cause parameter less than 80 hex shall not transmit another CTRL_CMD to that peer station for the duration of time designated in the LCR Cause parameter. |
| 5.2.2.2.29 | An LME receiving an LCR Cause parameter greater than 7F hex shall not transmit another CTRL_CMD to that peer system for the duration of time designated in the LCR Cause parameter. |
| NOTE: | A mobile LME receiving a station-based cause code from one ground station may immediately transmit the same CTRL_CMD to another ground station of the same ground system. |

Table 5.21: LCR cause parameter

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | LCR cause |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ | cause |
| | n + 4 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ | delay |
| | n + 5 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ | |
| | n + 6 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ | additional data |

Table 5.22: Cause code table

| Cause | Function | Additional data encoding | | | | | | | |
|--|---|----------------------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 00h | Bad local parameter. The additional data block, which may be repeated, contains the PI of a parameter which cannot be satisfied by this ground station. This cause will not be sent for an illegal Connection Management parameter. | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| | | p ₈ | p ₇ | p ₆ | p ₅ | p ₄ | p ₃ | p ₂ | p ₁ |
| 01h 02h 03h 04h 05h 06h 07 to 7Eh 7Fh | Out of link layer resources. Out of packet layer resources. Terrestrial network not available. Terrestrial network congestion. Cannot support autotune. Station cannot support initiating handoff. Reserved Other unspecified local reason. | undefined | | | | | | | |
| 80h | Bad global parameter. The additional data block, which may be repeated, contains the PI of a parameter which cannot be satisfied by any ground station in the system. This cause will not be sent for an illegal Connection Management Parameter. | identical to cause code 00 | | | | | | | |
| 81h | Protocol Violation. The first octet of the additional data block contains: 1) c/r bit (c bit) of the received CTRL; 2) re bit of the received CTRL; 3) Disconnected bit (d bit) shall be set to 1 if the LME has no links with the remote LME (the unexpected bit shall also be set to 1); 4) Illegal bit (i bit) shall be set to 1 if the LME receives an illegal CTRL (i.e. not listed in table 5.41 and described in clause 5.3.4); 5) Unexpected bit (u bit) shall be set to 1 if the LME receives a legal CTRL which is not legal in the context in which it was received. The remaining octets contains the parameter value of the Connection Management parameter (m bits) if included in the illegal CTRL. After transmitting or receiving an LCR with this cause Code, an LME shall delete all of its links. | 0 | 0 | 0 | u | i | d | p | c |
| | | m ₈ | m ₇ | m ₆ | m ₅ | m ₄ | m ₃ | m ₂ | m ₁ |
| 82h | Ground system out of resources. | | | | | | | | |
| 83 to FCh | Reserved. | | | | | | | | |
| FDh | Rejected for internal policy reasons. | | | | | | | | |
| | No response from VSS user. | | | | | | | | |
| FFh | Other unspecified system reason. | | | | | | | | |

| Requirement reference | |
|-----------------------|---|
| | UCTRL_DM parameter |
| 5.2.2.2.30 | The UCTRL_DM parameter shall define the period of time until an LME will accept new connections. |
| 5.2.2.2.31 | If the most significant bit is a one, then all existing connections shall be disconnected without any automatic recovery. |
| 5.2.2.2.32 | The seven least significant bits shall encode a field of 0 s to 127 s of time until new connections will be accepted. |
| 5.2.2.2.33 | If the UCTRL_DM parameter is included within a UCTRL DLPDU with the ucid subfield set to 4, then the UCTRL_DM parameter ID and parameter length shall be omitted and no other parameter including in the UCTRL. |
| 5.2.2.2.34 | The field shall be encoded per table 5.23. |

Table 5.23: UCTRL_DM parameter

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|--|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | |
| Parameter length | n + 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | |
| Parameter value | n + 3 | uda | nct ₇ | nct ₆ | nct ₅ | nct ₄ | nct ₃ | nct ₂ | nct ₁ | Bits 1 to 7: new connection timeout Bit 8: Disconnect All flag. |

5.2.2.3 Mobile-initiated information parameters

| Requirement reference | |
|-----------------------|---|
| 5.2.2.3.a | A mobile LME shall use mobile-initiated information parameters to inform the ground station about the capabilities or desires of that mobile. |
| 5.2.2.3.1 | Void. |
| | Modulation support parameter |
| 5.2.2.3.1a | This parameter shall define a list of modulation schemes that are supported. |
| 5.2.2.3.1b | This parameter shall be sent on link establishment. |
| 5.2.2.3.2 | The modulation support parameter shall be encoded as shown in tables 5.24 and 5.25. |

Table 5.24: Modulation support list encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Modulation support |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | res | res | res | res | m ₄ | m ₃ | m ₂ | m ₁ | Mode |

NOTE: 'res' refers to bits available for the information field.

Table 5.25: Modulation encoding

| Encoding | Meaning |
|----------|------------------------------|
| 0 to 1 | Reserved |
| 2 | VDL Mode 2 |
| 3 | VDL Mode 3 |
| 4 | VDL Mode 4 (GFSK modulation) |
| 5 to F | Reserved |

| Requirement reference | |
|-----------------------|--|
| | Acceptable alternate ground station parameter |
| 5.2.2.3.3 | The acceptable alternate ground station parameter shall define a list of ground stations in order of preference. |
| 5.2.2.3.4 | This parameter shall be defined by a list of DLS addresses decoded as per table 5.26. |
| 5.2.2.3.5 | These ground stations shall be used by the ground LME during handoffs as possible alternate ground stations, if the proposed ground station is not acceptable to the ground LME. |

Table 5.26: Acceptable alternate ground station parameter

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|--------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Alternate ground station |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | 0 | 0 | 0 | 0 | 0 | g ₂₇ | g ₂₆ | g ₂₅ | DLS Address |
| | n + 4 | g ₂₄ | g ₂₃ | g ₂₂ | g ₂₁ | g ₂₀ | g ₁₉ | g ₁₈ | g ₁₇ | |
| | n + 5 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ | |
| | n + 6 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ | |

| Requirement reference | |
|-----------------------|--|
| | Destination airport parameter |
| 5.2.2.3.6 | The destination airport parameter shall define the destination airport identifier of the mobile. |
| 5.2.2.3.7 | It shall be encoded as four 8-bit ISO IA-5 characters as defined in table 5.27. |

Table 5.27: Destination airport parameter

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Destination airport |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ | (first character) |
| | n + 4 | b ₈ | b ₇ | b ₆ | b ₅ | b ₄ | b ₃ | b ₂ | b ₁ | |
| | n + 5 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ | |
| | n + 6 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ | (fourth character) |

5.2.2.4 Ground-initiated modification parameters

| Requirement reference | |
|-----------------------|--|
| | Data link service parameter |
| 5.2.2.4.1 | The data link service parameter shall define the value of ND1, ND2, ND3, ND4, TD1, TD2 that a mobile is to use for priority levels Q1min to Q1max. |
| 5.2.2.4.2 | It shall be encoded as per table 5.28. |

Table 5.28: Data link service parameter encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|--------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | Data link service parameter |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | Q1max ₄ | Q1max ₃ | Q1max ₂ | Q1max ₁ | Q1min ₄ | Q1min ₃ | Q1min ₂ | Q1min ₁ | minimum priority (Q1min), maximum priority (Q1max) |
| | n + 4 | ND1 ₈ | ND1 ₇ | ND1 ₆ | ND1 ₅ | ND1 ₄ | ND1 ₃ | ND1 ₂ | ND1 ₁ | ND1 |
| | n + 5 | res | res | TD2 ₁₃ | TD1 ₁₃ | ND2 ₉ | ND1 ₁₁ | ND1 ₁₀ | ND1 ₉ | |
| | n + 6 | ND2 ₈ | ND2 ₇ | ND2 ₆ | ND2 ₅ | ND2 ₄ | ND2 ₃ | ND2 ₂ | ND2 ₁ | ND2 |
| | n + 7 | ND4 ₉ | ND3 ₇ | ND3 ₆ | ND3 ₅ | ND3 ₄ | ND3 ₃ | ND3 ₂ | ND3 ₁ | ND3 |
| | n + 8 | ND4 ₈ | ND4 ₇ | ND4 ₆ | ND4 ₅ | ND4 ₄ | ND4 ₃ | ND4 ₂ | ND4 ₁ | ND4 |
| | n + 9 | TD1 ₈ | TD1 ₇ | TD1 ₆ | TD1 ₅ | TD1 ₄ | TD1 ₃ | TD1 ₂ | TD1 ₁ | TD1 |
| | n + 10 | TD2 ₁₂ | TD2 ₁₁ | TD2 ₁₀ | TD2 ₉ | TD1 ₁₂ | TD1 ₁₁ | TD1 ₁₀ | TD1 ₉ | |
| | n + 11 | TD2 ₈ | TD2 ₇ | TD2 ₆ | TD2 ₅ | TD2 ₄ | TD2 ₃ | TD2 ₂ | TD2 ₁ | TD2 |

| Requirement reference | |
|-----------------------|---|
| | Autotune frequency parameter |
| 5.2.2.4.3 | This parameter shall define the frequency and modulation scheme that a mobile LME shall use to reply to a ground station listed in the replacement ground station list parameter. |
| 5.2.2.4.4 | This parameter shall be sent by a ground LME when an autotune is required. |
| 5.2.2.4.5 | The parameter shall be encoded as a 16-bit field as per table 5.29. |
| 5.2.2.4.6 | The modulation subfield (m bits) shall be defined as per table 5.25. |
| 5.2.2.4.7 | The frequency subfield (f bits) shall be defined as per ETSI EN 302 842-2 [6], table 5.26. |

Table 5.29: Autotune parameter encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|----------------|----------------|----------------|----------------|-----------------|-----------------|-----------------|----------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | Autotune frequency |
| Parameter length | n + 2 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | |
| Parameter value | n + 3 | m ₄ | m ₃ | m ₂ | m ₁ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ | |
| | n + 4 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ | |

| Requirement reference | |
|-----------------------|---|
| | Replacement ground station list parameter |
| 5.2.2.4.8 | The replacement ground station list parameter shall define a list of ground stations in order of ground LME preference. |
| 5.2.2.4.9 | This parameter shall be encoded as a list of DLS addresses in 32-bit fields as per table 5.30. |

Table 5.30: Replacement ground station list parameter encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | Replacement ground station list parameter |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | 0 | 0 | 0 | 0 | 0 | g ₂₇ | g ₂₆ | g ₂₅ | |
| | n + 4 | g ₂₄ | g ₂₃ | g ₂₂ | g ₂₁ | g ₂₀ | g ₁₉ | g ₁₈ | g ₁₇ | |
| | n + 5 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ | |
| | n + 6 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ | |

| Requirement reference | |
|-----------------------|---|
| | Re-transmission parameter |
| 5.2.2.4.10 | The re-transmission parameter shall define the value of Q5 that a mobile MAC uses for priority levels Q1min to Q1max. |
| 5.2.2.4.11 | The re-transmission parameter shall be encoded as per table 5.31. |

Table 5.31: Re-transmission parameter encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|--------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | Re-transmission parameter |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | Q1max ₄ | Q1max ₃ | Q1max ₂ | Q1max ₁ | Q1min ₄ | Q1min ₃ | Q1min ₂ | Q1min ₁ | minimum priority (Q1min), maximum priority (Q1max) |
| | n + 4 | min ₈ | min ₇ | min ₆ | min ₅ | min ₄ | min ₃ | min ₂ | min ₁ | Q5 _{min} |
| | n + 5 | 0 | min ₁₅ | min ₁₄ | min ₁₃ | min ₁₂ | min ₁₁ | min ₁₀ | min ₉ | |
| | n + 6 | max ₈ | max ₇ | max ₆ | max ₅ | max ₄ | max ₃ | max ₂ | max ₁ | Q5 _{max} |
| | n + 7 | 0 | max ₁₅ | max ₁₄ | max ₁₃ | max ₁₂ | max ₁₁ | max ₁₀ | max ₉ | |
| | n + 8 | mult ₈ | mult ₇ | mult ₆ | mult ₅ | mult ₄ | mult ₃ | mult ₂ | mult ₁ | Q5 _{mult} |
| | n + 9 | exp ₈ | exp ₇ | exp ₆ | exp ₅ | exp ₄ | exp ₃ | exp ₂ | exp ₁ | Q5 _{exp} |
| | n + 10 | 0 | 0 | 0 | 0 | num ₄ | num ₃ | num ₂ | num ₁ | Q5 _{num} |
| | n + 11 | 0 | wait ₇ | wait ₆ | wait ₅ | wait ₄ | wait ₃ | wait ₂ | wait ₁ | Q5 _{wait} |

| Requirement reference | |
|-----------------------|---|
| | Timer TL1 parameter |
| 5.2.2.4.12 | The timer TL1 parameter shall define the value of Timer TL1 (in seconds) that the initiating and responding LMEs use. |
| 5.2.2.4.13 | The Timer TL1 parameter shall be encoded as two 8-bit integers per table 5.32. |

Table 5.32: Timer TL1 parameter encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | Timer TL1 parameter |
| Parameter length | n + 2 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | |
| Parameter value | n + 3 | i ₈ | i ₇ | i ₆ | i ₅ | i ₄ | i ₃ | i ₂ | i ₁ | (initiating) |
| | n + 4 | r ₈ | r ₇ | r ₆ | r ₅ | r ₄ | r ₃ | r ₂ | r ₁ | (responding) |

| Requirement reference | |
|-----------------------|--|
| | Ground station address filter parameter |
| 5.2.2.4.14 | This parameter shall define the DLS address of the ground station from which links are handed-off. |
| 5.2.2.4.15 | This parameter shall be sent in a CTRL_CMD and the ground station address filter encoded as defined in table 5.33. |

Table 5.33: Ground station address filter parameter

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-------------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | Ground station address filter |
| Parameter length | n + 2 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | |
| Parameter value | n + 3 | 0 | 0 | 0 | 0 | 0 | g ₂₇ | g ₂₆ | g ₂₅ | DLS address |
| | n + 4 | g ₂₄ | g ₂₃ | g ₂₂ | g ₂₁ | g ₂₀ | g ₁₉ | g ₁₈ | g ₁₇ | |
| | n + 5 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ | |
| | n + 6 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ | |

| Requirement reference | |
|-----------------------|--|
| | Broadcast connection parameter |
| 5.2.2.4.16 | The broadcast connection parameter shall define a single mobile's link attributes for a new link, i.e.: the station address whose link was successfully established on the new link. |
| 5.2.2.4.17 | As per table 5.34, the mobile id subfield (a bits) shall be the station address. |

Table 5.34: Broadcast connection parameter encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | Ground station address filter parameter |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | 0 | 0 | 0 | 0 | 0 | a ₂₇ | a ₂₆ | a ₂₅ | mobile ID |
| | n + 4 | a ₂₄ | a ₂₃ | a ₂₂ | a ₂₁ | a ₂₀ | a ₁₉ | a ₁₈ | a ₁₇ | |
| | n + 5 | a ₁₆ | a ₁₅ | a ₁₄ | a ₁₃ | a ₁₂ | a ₁₁ | a ₁₀ | a ₉ | |
| | n + 6 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ | |

5.2.2.5 Ground-initiated information parameters

| Requirement reference | |
|-----------------------|---|
| | Airport coverage indication parameter |
| 5.2.2.5.1 | The airport coverage indication parameter shall define a list of four-character airport identifiers of airports for which the ground station can support communication with a mobile on the ground. |
| 5.2.2.5.2 | Each four-character identifier shall be encoded as four 8-bit ISO IA5 characters as per table 5.35. |

Table 5.35: Airport coverage indication parameter encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------------------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | Airport coverage indication parameter |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ | (first character) |
| | n + 4 | b ₈ | b ₇ | b ₆ | b ₅ | b ₄ | b ₃ | b ₂ | b ₁ | |
| | n + 5 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ | |
| | n + 6 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ | (fourth character) |

| Requirement reference | |
|-----------------------|---|
| | Nearest airport parameter |
| 5.2.2.5.3 | The nearest airport parameter shall define the four-character airport ID of the airport nearest the ground station. |
| 5.2.2.5.4 | It shall be encoded as four 8-bit ISO IA5 characters as per table 5.36. |
| 5.2.2.5.5 | The nearest airport parameter shall not be included in a CTRL if the Airport Coverage Indication is included. |

Table 5.36: Nearest airport parameter encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|---------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | Nearest airport parameter |
| Parameter length | n + 2 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | |
| Parameter value | n + 3 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ | (first character) |
| | n + 4 | b ₈ | b ₇ | b ₆ | b ₅ | b ₄ | b ₃ | b ₂ | b ₁ | |
| | n + 5 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ | |
| | n + 6 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ | (fourth character) |

| Requirement reference | |
|-----------------------|---|
| | ATN router NETs parameter |
| 5.2.2.5.6 | The ATN router NETs parameter shall define a list of ATN air-ground routers identified by the "administration identifier" (ADM) and "Administration Region Selector" (ARS) subfields of their network entity titles (NETs). |
| 5.2.2.5.7 | It shall be encoded as per table 5.37. |

Table 5.37: ATN router NETs parameter encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | ATN router NETs parameter |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | a ₂₄ | a ₂₃ | a ₂₂ | a ₂₁ | a ₂₀ | a ₁₉ | a ₁₈ | a ₁₇ | ADM subfield |
| | n + 4 | a ₁₆ | a ₁₅ | a ₁₄ | a ₁₃ | a ₁₂ | a ₁₁ | a ₁₀ | a ₉ | |
| | n + 5 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ | |
| | n + 6 | r ₂₄ | r ₂₃ | r ₂₂ | r ₂₁ | r ₂₀ | r ₁₉ | r ₁₈ | r ₁₇ | ARS subfield |
| | n + 7 | r ₁₆ | r ₁₅ | r ₁₄ | r ₁₃ | r ₁₂ | r ₁₁ | r ₁₀ | r ₉ | |
| | n + 8 | r ₈ | r ₇ | r ₆ | r ₅ | r ₄ | r ₃ | r ₂ | r ₁ | |

| Requirement reference | |
|-----------------------|---|
| | Ground-based system mask parameter |
| 5.2.2.5.8 | The ground-based system mask parameter shall define the ground-based system mask. |
| 5.2.2.5.9 | It shall be encoded as a 27-bit mask in a 32-bit field as per table 5.38. |

Table 5.38: Ground-based system mask parameter encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|------------------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | Ground-based system mask parameter |
| Parameter length | n + 2 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | |
| Parameter value | n + 3 | 0 | 0 | 0 | 0 | 0 | g ₂₇ | g ₂₆ | g ₂₅ | |
| | n + 4 | g ₂₄ | g ₂₃ | g ₂₂ | g ₂₁ | g ₂₀ | g ₁₉ | g ₁₈ | g ₁₇ | |
| | n + 5 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ | |
| | n + 6 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ | |

| Requirement reference | |
|--|--|
| | Frequency support list |
| 5.2.2.5.10 | The frequency support list, encoded per table 5.39, shall provide the mode, frequency, ground station address information and the relative location of the ground station listed. |
| 5.2.2.5.11 | The location shall be encoded as an offset relative to the position of the source station. |
| 5.2.2.5.12 | Easterly offset (eo) (West is negative) and Northerly offset (no) (South is negative) shall each be 8 bit parameters with a resolution of 4 NM encoded using two's complement mathematics. |
| 5.2.2.5.13 | The mode shall be encoded per table 5.25. |
| 5.2.2.5.14 | The frequency shall be encoded per ETSI EN 302 842-2 [6], table 5.25. |
| NOTE: The usage of this parameter is defined in clause 5.2.5a. | |

Table 5.39: Frequency support list encoding

| Field | Octet | Bit number | | | | | | | | Notes |
|------------------|--------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-------------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Parameter ID | n + 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | frequency support list |
| Parameter length | n + 2 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | |
| Parameter value | n + 3 | m ₄ | m ₃ | m ₂ | m ₁ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ | mode/frequency |
| | n + 4 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ | |
| | n + 5 | 0 | 0 | 0 | 0 | 0 | g ₂₇ | g ₂₆ | g ₂₅ | ground station identification |
| | n + 6 | g ₂₄ | g ₂₃ | g ₂₂ | g ₂₁ | g ₂₀ | g ₁₉ | g ₁₈ | g ₁₇ | |
| | n + 7 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ | |
| | n + 8 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ | |
| | n + 9 | eo ₈ | eo ₇ | eo ₆ | eo ₅ | eo ₄ | eo ₃ | eo ₂ | eo ₁ | easterly offset |
| | n + 10 | no ₈ | no ₇ | no ₆ | no ₅ | no ₄ | no ₃ | no ₂ | no ₁ | northerly offset |

5.2.3 LME timers and parameters

5.2.3.1 General

| Requirement reference | |
|-----------------------|--|
| 5.2.3.1.1 | The LME service shall implement the system parameters defined in table 5.40. |

Table 5.40: Management entity system parameters

| Symbol | Parameter name | Minimum | Maximum | Default | Increment |
|--------|--------------------------------|---------|---------|---------|-----------|
| TL1 | Maximum link overlap time | | | | |
| | initiating | 0 s | 255 s | 20 s | 1 s |
| | responding | 0 s | 255 s | 60 s | 1 s |
| TL2 | Link initialization time | 5 s | 25 s | 6 s | 1 ms |
| TL4 | Leave event generation latency | 0 s | 255 s | 20 s | 1 s |

5.2.3.2 Timer TL1 (maximum link overlap time)

| Requirement reference | |
|---------------------------------------|--|
| 5.2.3.2.1 | Timer TL1 shall be set to the maximum time that initiating and responding LMEs will maintain the old link during handoffs. |
| 5.2.3.2.2 | The LME initiating the handoff shall start its Timer TL1 when it receives an CTRL_RSP_HO. |
| 5.2.3.2.3 | The LME responding to the handoff shall start its Timer TL1 when it transmits its CTRL_RSP_HO. |
| 5.2.3.2.4 | The initiating LME shall never restart its Timer TL1. |
| 5.2.3.2.5 | The responding LME shall restart its Timer TL1 if it retransmits a CTRL_RSP_HO. |
| 5.2.3.2.6 | Timer TL1 shall be cancelled if either the old or new link is prematurely disconnected. |
| 5.2.3.2.7 | After TL1 expires, each LME shall silently disconnect its half of the old link. |
| NOTE: There is one timer TL1 per LME. | |

5.2.3.3 Parameters TL2 (link initialization time)

| Requirement reference | |
|-----------------------|---|
| 5.2.3.3.1 | The parameter TL2 shall control the retransmission of CTRL DLPDU for which an expected response has not been received. |
| 5.2.3.3.2 | Timer TL2 shall be set after the transmission of a CTRL for which a response is expected. |
| 5.2.3.3.3 | Timer TL2 shall be cleared upon receipt of a CTRL DLPDU from the peer LME. |
| 5.2.3.3.4 | If Timer TL2 expires, the LME shall attempt to handoff to another ground station in accordance with the procedures of clause 5.2.5. |

5.2.3.4 Timer TL4 (leave generation latency)

| Requirement reference | |
|---|--|
| 5.2.3.4.1 | Timer TL4 shall control the triggering of Leave events. |
| 5.2.3.4.2 | Timer TL4 shall be set whenever the LME initiates a handoff because of an invalid link. |
| 5.2.3.4.3 | Timer TL4 shall be cancelled upon completion of a successful handoff. |
| 5.2.3.4.4 | If Timer TL4 expires, the LME shall generate a Leave event as described in clause 5.2.6. |
| 5.2.3.4.5 | Timer TL4 shall never be restarted. |
| NOTE 1: There is one TL4 timer per peer station. | |
| NOTE 2: Instances of when a link may be considered invalid are upon expiration of Q5num or Q5wait, upon exceedance of L1 and upon receipt of a DM/DISC or a UCTRL_DM. | |
| NOTE 3: When situated in the centre of the service volume the generation of the leave event should be suppressed for a period of time on the assumption that the datalink will become available again, thus the value of TL4 should provide sufficient time to complete a handoff. When on the edge of the service volume, a Leave event should be generated as soon as the unreachability is declared internally, thus the value of TL4 should be low. | |

5.2.4 CTRL DLPDU types and procedures

| Requirement reference | |
|-----------------------|--|
| 5.2.4.1 | The mobile and ground LMEs shall use the CTRL DLPDU types listed in table 5.41 and the procedures described below to provide a reliable connection between the mobile and ground-based system. |
| 5.2.4.2 | If an LME receives any valid CTRL_HO frame from a system with which it does not have a link, it shall respond with a CTRL_LCR with the "d" bit set to 1 in the Protocol Violation Cause Code. |

Table 5.41: CTRL parameters

| | GSIF | | Air initiated link establishment | |
|---|---------------------|----------------|----------------------------------|--------------------|
| | Source address | Ground station | Mobile | New ground station |
| | Destination address | All aircraft | Proposed ground station | Mobile |
| CTRL parameters | PI hex | GSIF | CTRL_CMD_LE | CTRL_RSP_LE |
| Connection management | 01 hex | N/A | M | M |
| CTRL sequencing | 02 hex | N/A | M | M |
| Protocol options | 03 hex | N/A | O | O |
| LCR cause | 04 hex | N/A | N/A | N/A |
| UCTRL_DM | 05 hex | O | N/A | N/A |
| Datalink service | 40 hex | O | N/A | O |
| VSS sublayer | 41 hex | O | N/A | O |
| Quality of Service | 42 hex | O | N/A | O |
| m2 filter | 43 hex | O | N/A | O |
| CG1 filter | 44 hex | O | N/A | O |
| Autotune | 45 hex | O | N/A | O |
| Maximum number of missed reservations | 46 hex | N/A | N/A | O |
| Repl. Ground station | 47 hex | O | N/A | O |
| Random access | 48 hex | O | N/A | O |
| Re-transmission | 49 hex | O | N/A | O |
| Timer TL1 | 4A hex | O | N/A | O |
| Ground Station Address filter | 4B hex | N/A | N/A | N/A |
| Broadcast connection | 4C hex | N/A | N/A | N/A |
| Modulation support | 80 hex | N/A | O | N/A |
| Alternate grd stations | 81 hex | N/A | O | N/A |
| Destination airport | 82 hex | N/A | O | N/A |
| Airport coverage | C0 hex | O | N/A | O |
| Nearest airport | C1 hex | M ¹ | N/A | O ² |
| ATN router NETs | C2 hex | M | N/A | O |
| Ground-based system mask | C3 hex | M | N/A | M |
| Frequency support | C4 hex | O | N/A | O |
| Directory of service | C5 hex | O | N/A | O |
| Channel management | C6 hex | O | N/A | O |
| PI = Parameter identifier. M = Mandatory. O = Optional. N/A = Not applicable. hex = hexadecimal. | | | | |
| NOTE 1: In a GSIF DLPDU it is mandatory to include either the Airport Coverage Indication parameter or the Nearest Airport Identifier parameter but not both. | | | | |
| NOTE 2: Where the Airport Coverage Indication parameter and the Nearest Airport Identifier parameter are marked as optional, either parameter may be included in the DLPDU or neither but not both. | | | | |

Table 5.41 (continued): CTRL parameters

| | Ground initiated handoff | | | Air initiated handoff | |
|---|--------------------------|-------------------------|-----------------------|-------------------------|-----------------------|
| | Source address | Proposed ground station | Mobile | Mobile | New ground station |
| | Destination address | Mobile | New ground station | Proposed ground station | Mobile |
| CTRL parameters | PI hex | CTRL_CMD_HO (re=1) | CTRL_RSP_HO (re=1) | CTRL_CMD_HO (re=1) | CTRL_RSP_HO (re=1) |
| Connection Management | 01h | M | M | M | M |
| CTRL sequencing | 02h | M | M | M | M |
| Protocol options | 03h | O | O | O | O |
| LCR cause | 04h | N/A | N/A | N/A | N/A |
| UCTRL_DM | 05h | N/A | N/A | N/A | N/A |
| Datalink service | 40h | O | N/A | N/A | O |
| VSS sublayer | 41h | O | N/A | N/A | O |
| Quality of service | 42h | O | N/A | N/A | O |
| m2 filter | 43h | O | N/A | N/A | O |
| CG1 filter | 44h | O | N/A | N/A | O |
| Autotune | 45h | O | N/A | N/A | O |
| Maximum number of missed reservations | 46h | O | N/A | N/A | O |
| Repl. Ground station | 47h | O | N/A | N/A | O |
| Random access | 48h | O | N/A | N/A | O |
| Re-transmission | 49h | O | N/A | N/A | O |
| Timer TL1 | 4Ah | O | N/A | N/A | O |
| Ground Station Address filter | 4Bh | N/A | N/A | N/A | N/A |
| Broadcast connection | 4Ch | N/A | N/A | N/A | N/A |
| Modulation support | 80h | N/A | N/A | N/A | N/A |
| Alternate ground stations | 81h | N/A | N/A | O | N/A |
| Destination airport | 82h | N/A | O | O | N/A |
| Airport coverage | C0h | O ¹ | N/A | N/A | O ¹ |
| Nearest airport ID | C1h | O ¹ | N/A | N/A | O ¹ |
| ATN router NETs | C2h | M | N/A | N/A | O |
| Ground-based system mask | C3h | M | N/A | N/A | O |
| Frequency support | C4h | O | N/A | N/A | O |
| Directory of service | C5h | O | N/A | N/A | O |
| Channel management | C6h | O | N/A | N/A | O |
| PI = Parameter identifier. M = Mandatory. O = Optional. N/A = Not applicable. h = hexadecimal. | | | | | |
| NOTE: Where the Airport Coverage Indication parameter and the Nearest Airport Identifier parameter are marked as optional, either parameter may be included in the DLPDU or neither but not both. | | | | | |

Table 5.41 (concluded): CTRL parameters

| | | Air requested handoff | Ground requested handoff | Ground requested broadcast | Link connection rejection |
|---------------------------------------|---------------------|------------------------------------|--------------------------|----------------------------|------------------------------|
| | Source address | Mobile | Current ground station | New ground station | Any station |
| | Destination address | Current or proposed ground station | Mobile | All mobiles | Any station |
| CTRL parameters | PI hex | CTRL_CMD_HO (re=0) | CTRL_CMD_HO (re=0) | CTRL_CMD_HO (re=0) | CTRL_RSP_LCR CTRL_CMD_LCR |
| Connection management | 01h | M | M | M | M |
| CTRL sequencing | 02h | M | M | M | M |
| Protocol options | 03h | O | O | O | O |
| LCR cause | 04h | N/A | N/A | N/A | M |
| UCTRL_DM | 05h | N/A | N/A | N/A | N/A |
| Data link service | 40h | N/A | O | O | N/A |
| VSS sublayer | 41h | N/A | O | O | N/A |
| Quality of Service | 42h | N/A | O | O | N/A |
| m2 filter | 43h | N/A | O | O | N/A |
| CG1 filter | 44h | N/A | O | O | N/A |
| Autotune | 45h | N/A | O | O | N/A |
| Maximum number of missed reservations | 46h | N/A | O | O | N/A |
| Repl. Ground station | 47h | N/A | O | O | N/A |
| Random access | 48h | N/A | O | O | N/A |
| Re-transmission | 49h | N/A | O | O | N/A |
| Timer TL1 | 4Ah | N/A | O | O | N/A |
| Ground Station Address filter | 4Bh | N/A | N/A | M | N/A |
| Broadcast connection | 4Ch | N/A | N/A | M | N/A |
| Modulation support | 80h | N/A | N/A | O | N/A |
| Alternate ground | 81h | O | N/A | N/A | N/A |
| Destination airport | 82h | O | N/A | N/A | N/A |
| Airport coverage | C0h | N/A | N/A | O ¹ | N/A |
| Nearest airport | C1h | N/A | N/A | O ¹ | N/A |
| ATN router NETs | C2h | N/A | O | M | N/A |
| Ground-based system mask | C3h | N/A | O | O | N/A |
| Frequency support | C4h | N/A | O | O | N/A |
| Directory of service | C5h | N/A | O | O | N/A |
| Channel management | C6h | N/A | O | O | N/A |

PI = Parameter identifier.

M = Mandatory.

O = Optional.

N/A = Not applicable.

h = hexadecimal.

NOTE: Where the Airport Coverage Indication parameter and the Nearest Airport Identifier parameter are marked as optional, either parameter may be included in the DLPDU or neither but not both.

5.2.5 CTRL transmission procedures

5.2.5a Frequency management procedures

| Requirement reference | |
|-----------------------|---|
| 5.2.5.a.1 | The mobile LME shall use the following procedures to acquire a frequency on which reliable VDL services are available. |
| | Frequency search |
| 5.2.5.a.2 | The mobile LME shall initiate the frequency search procedure on system initialization or after link disconnection, if there are no ground stations providing the requested service which are reachable according to the definition of clause 5.2.3.2. |
| 5.2.5.a.3 | It shall attempt to identify a frequency on which the required VDL service is available by tuning the radio to the GSCs and/or to other frequencies on which it knows a-priori that VDL service is available. |
| 5.2.5.a.4 | On the basis of information contained in the GSIF, it shall select an appropriate channel to receive the required service. |
| | Frequency recovery |
| 5.2.5.a.5 | The mobile LME shall initiate the frequency recovery procedure if it can no longer establish a link on the current frequency or if the VSS indicates that the current frequency is congested. |
| 5.2.5.a.6 | It shall tune the radio to an alternate frequency using the data in the Frequency Support List or Directory of Service message previously received on the current link. |

5.2.5.1 Link connectivity procedures

| Requirement reference | |
|-----------------------|---|
| 5.2.5.1.1 | <p>The mobile LME shall use the following procedures to maintain connectivity across the VHF link:</p> <ul style="list-style-type: none"> a) ground station identification; b) initial link establishment; c) mobile-initiated handoff; d) mobile-requested ground-initiated handoff; e) ground-initiated handoff; f) ground-requested mobile-initiated handoff; g) ground-requested mobile-initiated handoff; h) autotune. |

5.2.5.2 Ground Station Identification

| Requirement reference | |
|-----------------------|---|
| 5.2.5.2.1 | Void. |
| 5.2.5.2.2 | Void. |
| 5.2.5.2.3 | Mobile LMEs receiving a GSIF shall process its content to identify the functionality of the ground station as well as the correct operational parameters to be used when communicating with it. |
| 5.2.5.2.4 | Mobile LMEs which have a connection with the transmitting ground station shall process only information parameters as per table 5.41. |

5.2.5.3 Link establishment

| Requirement reference | |
|--------------------------------|---|
| 5.2.5.3.a | The mobile LME shall initiate the link establishment procedure with a ground station to establish an initial link with the ground-based system. |
| 5.2.5.3.b | Whenever the link is disconnected (e.g. on receipt of DM/DISC or DM/FRMR), a mobile shall initiate link establishment according to the local link management policy if no links remain. |
| Mobile initiation | |
| 5.2.5.3.c | The mobile LME shall choose a ground station with which it wishes to establish a link based on its capability to support a link and so as to maximize the likely duration of the connection to the ground station (see note). |
| 5.2.5.3.d | It shall then attempt to establish a link with the chosen ground station by sending a CTRL_CMD_LE (re=1) DLPDU. |
| 5.2.5.3.e | This DLPDU shall include the mandatory parameters as per table 5.41 and also any optional parameters for which the mobile LME does not wish to use the default value. |
| 5.2.5.3.f | If the mobile LME has received a GSIF from the ground station to which it is transmitting the CTRL_CMD_LE (re=1), then it shall use the parameters as declared; otherwise, it shall use the default parameters. |
| General ground response | |
| 5.2.5.3.1 | Void. |
| 5.2.5.3.2 | Void. |
| 5.2.5.3.3 | Void. |
| 5.2.5.3.4 | Void. |
| Exceptional cases | |
| 5.2.5.3.5 | Void. |
| 5.2.5.3.6 | If the parameters in the CTRL_RSP_LE from the ground LME are not acceptable to the mobile LME, then the mobile LME shall transmit a DM/DISC to the ground. |
| 5.2.5.3.7 | If the Autotune parameter is included in the CTRL_RSP_LE and the mobile LME is unable to perform the autotune, then the mobile LME shall respond with an CTRL_CMD_LCR (re=0); the link established on the current frequency will not be affected. |
| NOTE: | To maximize the likely connection time, the mobile can take account of mobile position, intent, ground station position, the signal quality of all received uplink bursts and on information in any received GSIFs. |

5.2.5.4 Mobile-initiated handoff

| Requirement reference | |
|-----------------------|---|
| 5.2.5.4.a | If a mobile LME implements this clause, then it shall set the "i" bit in the Protocol Options parameter to 1; otherwise, it shall set the "i" bit to 0. |
| | Mobile handoff |
| 5.2.5.4.b | Once the mobile LME has established a link to a ground station, it shall monitor the VHF signal quality on the link and the transmissions of the other ground stations. |
| 5.2.5.4.c | The mobile LME shall establish a link to a new ground station if any of the following events occur: <ul style="list-style-type: none"> a) the VHF signal quality on the current link is determined, in accordance with local link management policy, to be insufficient to maintain reliable communications and the signal quality of another ground station is significantly better; b) TL2 seconds have elapsed since the LME initiated the request to send any burst to the current ground station; c) the peer station has become unreachable as defined in clause 5.2.3.2; d) Timer TM2 expires; e) the mobile is at a position which, in accordance with local link management policy, requires the station to establish a link with a new ground station. |
| 5.2.5.4.d | If timer TM2 expires, the mobile LME shall autonomously tune to an alternate frequency (provided in a frequency support list) before initiating the handoff. |
| 5.2.5.4.e | A mobile that wishes to initiate a handoff to a new ground station shall send to that ground station a CTRL_CMD_HO (re = 1) with parameters as defined in table 5.41. |
| | Site selection preference |
| 5.2.5.4.f | From among those ground stations with acceptable link quality, the mobile LME shall prefer to handoff to a ground station which indicates (in the GSIF) accessibility to the air-ground router(s) to which the mobile DTE has subnetwork connections. |
| | Recommendation |
| 5.2.5.4.g | If a mobile has commenced approach to its destination airport and its current link is with a ground station that does not offer service at that airport, it should handoff to a ground station which indicates in its Airport Coverage Indication parameter that it offers service at that airport. |
| | Interaction of LMEs |
| 5.2.5.4.1 | When a mobile VME hands off from a ground station in one ground-based system (and thus associated with one LME) to a ground station in another ground-based system (and thus associated with a different LME in the mobile), the new LME shall use the link establishment procedures. |
| 5.2.5.4.2 | The old LME shall send a DM/DISC when directed by the VME. (See note) |
| | General ground response |
| 5.2.5.4.3 | Void. |
| 5.2.5.4.4 | Void. |
| 5.2.5.4.5 | Void. |
| 5.2.5.4.6 | Void. |
| | Disconnecting old link |
| 5.2.5.4.7 | If the new and old ground stations are associated with different systems, then the procedures of clauses 5.2.5.4.1 and 5.2.5.4.2 shall be followed. |
| 5.2.5.4.8 | Otherwise, the mobile LME shall set Timer TL1 when it receives the CTRL_RSP_HO from the ground LME. |
| 5.2.5.4.9 | The ground LME shall set Timer TL1 after it transmits the CTRL_RSP_HO. |
| 5.2.5.4.10 | Both stations shall continue to operate on the old link until their respective Timer TL1 expires, after which each will consider the link disconnected without sending or receiving a DM/DISC. |
| | Exceptional cases |
| 5.2.5.4.11 | Void. |
| 5.2.5.4.12 | Void. |
| 5.2.5.4.13 | If more than TL2 seconds have elapsed since the LME initiated the request to send the CTRL_CMD_HO (re = 1), the aircraft LME shall attempt to handoff to another ground station; the current link will not be affected. |
| 5.2.5.4.14 | If the mobile LME cannot perform the autotune, it shall transmit a CTRL_CMD_LCR (re=0); the current link will not be affected. |
| 5.2.5.4.15 | If the parameters in the CTRL_RSP_HO are not acceptable to the mobile LME, then the mobile LME shall transmit a DM/DISC to the ground on the new link. |
| NOTE: | Optimally the old link should not be disconnected until after the new link is capable of carrying application data. This subject is however outside the scope of the present document. |

5.2.5.5 Mobile-requested ground-initiated handoff

| Requirement reference | |
|-----------------------|---|
| 5.2.5.5.a | A mobile LME shall not perform this clause when its peer LME does not support handoff initiation. |
| 5.2.5.5.b | A mobile LME shall only perform this clause if the current and proposed ground stations are both managed by its peer LME. |
| | Mobile action |
| 5.2.5.5.c | For a mobile LME to request the ground LME to initiate a handoff, it shall send an CTRL_CMD_HO (re=0) addressed to its current or proposed ground station with the parameters as per table 5.41. |
| 5.2.5.5.d | During this procedure the current link shall not be affected until the mobile LME receives a CTRL_CMD_HO (re=1). |
| | General ground response |
| 5.2.5.5.1 | Void. |
| 5.2.5.5.2 | Void. |
| | Exceptional cases |
| 5.2.5.5.3 | Void. |
| 5.2.5.5.4 | Void. |
| | Recommendation |
| 5.2.5.5.5 | If more than TL2 seconds have elapsed since the LME initiated the request to send the CTRL_CMD_HO, the mobile LME should attempt to request to handoff to another station before disconnecting all links to the ground and restarting link establishment. |

5.2.5.6 Ground-initiated handoff

| Requirement reference | |
|-----------------------|---|
| 5.2.5.6.1 | Void. |
| 5.2.5.6.2 | Void. |
| | Ground action |
| 5.2.5.6.3 | Void. |
| 5.2.5.6.4 | Void. |
| 5.2.5.6.5 | Any operating parameters in the CTRL_CMD_HO (re = 1) transmitted by the ground station (either modification or informational) shall be valid for the transmitting station and for all ground stations listed in the Replacement Ground Station List parameter, except the Airport Coverage Indication parameter and Nearest Airport parameter which are only valid for the transmitting ground station. |
| | General mobile response |
| 5.2.5.6.5a | The mobile LME shall respond by sending a CTRL_RSP_HO with parameters as per table 5.41 to either the proposed ground station or to its preferred ground station if the CTRL_CMD_HO included the Replacement Ground Station List parameter. |
| | Disconnecting old link |
| 5.2.5.6.5b | The mobile LME shall set Timer TL1 after it transmits the CTRL_RSP_HO. |
| 5.2.5.6.6 | Void. |
| 5.2.5.6.7 | Although new traffic will be sent over the new link, the old link shall not be disconnected immediately to allow any old traffic to be delivered. |
| | Exceptional cases |
| 5.2.5.6.7a | If the mobile LME cannot accept the handoff request, it shall respond with a CTRL_RSP_LCR; the current link will not be affected. |
| 5.2.5.6.8 | Void. |
| | Recommendation |
| 5.2.5.6.9 | Void. |

5.2.5.7 Ground-requested mobile-initiated handoff

| Requirement reference | |
|--------------------------------|--|
| 5.2.5.7.1 | Void. |
| Ground action | |
| 5.2.5.7.2 | Void. |
| 5.2.5.7.3 | Void. |
| 5.2.5.7.4 | Void. |
| 5.2.5.7.5 | Void. |
| General mobile response | |
| 5.2.5.7.5a | If the mobile LME receives a CTRL_CMD_HO (re = 0), it shall commence an aircraft-initiated handoff to a ground station, preferably one listed in the Replacement Ground Station List parameter. |
| Exceptional cases | |
| 5.2.5.7.5b | If the mobile LME cannot initiate the handoff, it shall send a CTRL_CMD_LCR (re = 0); the current link will not be affected. |
| 5.2.5.7.5c | If the Autotune parameter is included in the CTRL_CMD_HO (re = 0), the mobile LME shall transmit the CTRL_CMD_HO (re = 1) on the new frequency, and retransmit on the new frequency, using the normal retransmission procedures, if no response is obtained. |
| 5.2.5.7.5d | Otherwise, it shall only transmit the CTRL_CMD_HO (re = 1) once per received CTRL_CMD_HO (re = 0). |
| Recommendation | |
| 5.2.5.7.6 | Void. |

5.2.5.8 Ground-requested broadcast handoff

| Requirement reference | |
|--------------------------|---|
| 5.2.5.8.1 | Void. |
| 5.2.5.8.2 | Void. |
| Ground action | |
| 5.2.5.8.3 | Void. |
| Mobile response | |
| 5.2.5.8.4 | The LME in each mobile shall process received broadcast CTRL_CMD_HO (re=0) and determine if the ground LME had performed a broadcast link recovery for it. |
| 5.2.5.8.5 | It shall do this by verifying that the Ground Station Address Filter parameter contains the DLS address of the ground station that it is connected to and that a Broadcast Connection parameter exists containing its station address. |
| 5.2.5.8.6 | Mobile LMEs supporting broadcast recovery shall consider that a link handoff has occurred with the new link having the same parameters as the old link (as modified by the parameters in the broadcast CTRL). The old link shall be disconnected immediately. |
| Exceptional cases | |
| 5.2.5.8.7 | If the mobile LME does not support broadcast recovery, but the ground LME performed a broadcast link recovery for it, then the mobile LME shall perform either an air-initiated link handoff, (if the aircraft LME supports same), or request a link handoff. |
| 5.2.5.8.8 | If the mobile LME finds the new ground station unacceptable, it shall perform an air-initiated handoff (if the aircraft LME supports same), or request a link handoff. |
| 5.2.5.8.9 | If the Ground Station Address Filter parameter does not equal the DLS address of a link that the mobile LME has or if no mobile identifier subfield in a Broadcast Connection parameter equals its mobile address, the mobile LME shall not process the ground requested broadcast handoff. |

5.2.5.9 Ground-commanded autotune

| Requirement reference | |
|-----------------------|--|
| | Ground action |
| 5.2.5.9.1 | Void. |
| | General mobile response |
| 5.2.5.9.2 | On receipt of a CTRL commanding an autotune, the mobile LME shall retune the mobile radio to the new frequency and commence a mobile-initiated handoff to the chosen ground station. |
| | Exceptional cases |
| 5.2.5.9.3 | If the mobile LME cannot perform the autotune, it shall transmit a CTRL_CMD_LCR (re=0); the current link shall not be affected. |

5.2.6 VDL Mode 4 Mobile SubNetwork Dependent Convergence Function (SND CF)

5.2.6.1 Frame mode SND CF

| Requirement reference | |
|---|--|
| 5.2.6.1.1 | The VDL Mode 4 shall support the Frame Mode SND CF as defined in ICAO ATN SARPs [3], clause 5.7.8. |
| 5.2.6.1.2 | To support the interface, VDL Mode 4 shall generate a Join event upon reception of a CTRL_RSP_LE or a CTRL_RSP_HO; |
| 5.2.6.1.3 | and shall generate a Leave event whenever either the link is terminated, or the link is abandoned; |
| 5.2.6.1.4 | and shall provide a data transfer service as defined in clause 5.1. |
| NOTE 1: A link can be terminated for a variety of reasons, including a station becoming unreachable, protocol exchanges and link time-outs (e.g. expiration of the TL1 or the TL2 timers). A link is abandoned when the TM2 timer indicates that the frequency is congested, the LME enters frequency recovery mode trying to find a less congested frequency and the VDL4 station does not have a spare receiver to monitor both the old and new channels. | |
| NOTE 2: The timeliness of the generation of LEAVE events depends on the value of the L1 counter and the number of reservations known to have been made by a station within a time interval. | |

5.3 Additional VSS requirements

NOTE: The VSS requirements to support core link layer functionality are contained in ETSI EN 302 842-2 [6]. The requirements defined here are additional requirements to support the DLS.

5.3.1 Information transfer request protocol specification

NOTE: This protocol is intended for a VSS user which requires a peer VSS user to send a response of length, lg. The protocol also allows the requesting VSS user to place a reservation for an acknowledgement by the requesting VSS user to the response field.

5.3.1.1 Information transfer request parameters

| Requirement reference | |
|-----------------------|--|
| 5.3.1.1.1 | The information transfer request protocol shall implement the system parameters defined in table 5.42. |

Table 5.42: Information transfer request VSS system parameters

| Symbol | Parameter name | Minimum | Maximum | Recommended default | Increment |
|--------|------------------------------------|---------|-------------|--|-----------|
| V42 | Length of information transfer | 1 slot | 256 slots | N/A (depends on information to be transmitted) | 1 slot |
| V43 | Minimum information transfer delay | 2 slots | 500 slots | 20 slots | 1 slot |
| V44 | Maximum information transfer delay | 2 slots | 2 047 slots | 1 000 slots | 1 slot |
| V45 | Minimum response delay | 2 slots | 500 slots | 20 slots | 1 slot |
| V46 | Maximum response delay | 2 slots | 2 047 slots | 1 000 slots | 1 slot |

| Requirement reference | |
|--|---|
| 5.3.1.1.2 | The VSS user shall provide the destination address and any of the parameters V42, V43, V44, V45, V46 and quality of service parameters (Q2a to Q2d and Q4) for which the default values are not desired. |
| | Parameter V42 (length of information transfer) |
| 5.3.1.1.3 | Parameter V42 shall be the number of slots required for information transfer. |
| | Parameter V43 (minimum information transfer delay) |
| 5.3.1.1.4 | Parameter V43 shall be the minimum delay, measured in slot intervals, that a station will provide to a responder in order to ensure that the responder can generate the required information for transfer before its reserved slots (see note 1). |
| | Parameter V44 (maximum information transfer delay) |
| 5.3.1.1.5 | Parameter V44 shall be the maximum delay, measured in slot intervals, that a station will provide to a responder in order to ensure timely delivery in case a retransmission is required. |
| | Parameter V45 (minimum response delay) |
| 5.3.1.1.6 | Parameter V45 shall be the minimum delay, measured in slot intervals after the information transfer that the requesting station will require in order to generate an acknowledgement to the information transfer in order to ensure that the requesting station can generate the acknowledgement before its reserved slot (see note 2). |
| | Parameter V46 (maximum response delay) |
| 5.3.1.1.7 | Parameter V46 shall be the maximum delay, measured in slot intervals after the information transfer that the requesting station will require in order to ensure timely delivery of the acknowledgement in case a retransmission is required. |
| NOTE 1: $V43 \times 60/M1$ is the maximum time that a station is provided with to generate a response to the request. | |
| NOTE 2: $V45 \times 60/M1$ is the maximum time that a station is provided with to generate an acknowledgement to the information transfer. | |

5.3.1.2 Information transfer request transmission procedures

| Requirement reference | |
|-----------------------|--|
| | Selection of the transmission slot for the information transfer request reservation |
| 5.3.1.2.1 | If no slot has been reserved for transmission of an information transfer request reservation, the station shall select a slot using the random access procedures (see ETSI EN 302 842-2 [6], clause 5.2.7). |
| 5.3.1.2.2 | The transmission slot (t_slot) shall be the slot containing the information transfer request reservation transmission. |
| | Selection of the reserved slots for the response |
| 5.3.1.2.3 | A block of slots of length V42 to be reserved for the response shall be selected using the slot selection procedure specified in ETSI EN 302 842-2 [6], clause 5.2.6.2, using VSS user supplied quality of service parameters, and candidate slots in the range V43 to V44 after the transmitted burst. |
| 5.3.1.2.4 | The reserved slot (r_slot) shall be the chosen slot or the first slot in the chosen group of slots. |
| | Selection of the reserved slot for the acknowledgement |
| 5.3.1.2.5 | The acknowledgement slot (a_slot) shall be selected using the slot selection procedure specified in ETSI EN 302 842-2 [6], clause 5.2.6.2, using VSS user supplied quality of service parameters, and candidate slots in the range V45 to V46 after the end of the slot or group of slots reserved for the response. |
| | Information transfer request burst transmission |
| 5.3.1.2.6 | A station sending an information transfer request burst to its peer shall include the information transfer request reservation field. |
| 5.3.1.2.7 | It shall set the destination (d) subfield to the destination of the burst, the response offset (ro) subfield to a value of (r_slot - t_slot - 1), the length (lg) subfield equal to (V42-1), the frequency (f) subfield set to the channel on which information transfer is required and the acknowledgement offset (ao) subfield set to a value of (a_slot - r_slot - lg - 1). |
| | Action after no response |
| 5.3.1.2.8 | If a response is not received by the reserved information transfer slots, then the station shall inform the VSS user that no response has been received and transmit a General Failure with cause code 7E hex or FE hex (see ETSI EN 302 842-2 [6], clause 5.2.20.2.1) or a NACK (see clauses 5.1.4.5.22 to 5.1.4.5.25) in the slot reserved for the acknowledgement (see note). |
| NOTE: | If the information transfer protocol is being used as part of DLS long transmission procedure, the defined action is to send a NACK in the slot reserved for the acknowledgement. |

5.3.1.3 Information transfer request acknowledgement procedures

| Requirement reference | |
|-----------------------|---|
| 5.3.1.3.1 | The acknowledgement shall be on the same frequency as the information transfer reservation burst that was used to reserve a slot for the acknowledgement. |

5.3.2 Void

5.3.3 Void

5.3.4 Retransmission procedures

| Requirement reference | |
|--|---|
| 5.3.4.1 | After transmitting a burst containing a reservation for a peer station (i.e. unicast request reservation, directed request reservation, information transfer request reservation) and not receiving a response by the expected slot, a station shall either retransmit the request or inform the VSS user and LME if Q5num attempts have already been made or if more than Q5wait seconds have elapsed since the VSS user initiated the request. |
| 5.3.4.2 | The re-transmitting station shall wait for $Q5min + \min(U(x), Q5max)$ seconds before attempting to retransmit the burst, where: <ul style="list-style-type: none"> - U(x) is a uniform random number generated between 0 and x; - x is defined by $Q5mult \times (Q5exp^{retrans}) \times M1 / (M1 + 1 - u)$; - u is the number of occupied slots within the past minute on the channel concerned; - retrans is the number of times that a burst has been retransmitted (see note). |
| NOTE: If Q5num = 1, no re-transmission is attempted and hence parameters Q5max, Q5min, Q5mult, Q5exp are not used. | |

6 General design requirements

The equipment shall meet the requirements of ETSI EN 302 842-2 [6], clause 6.

7 Protocol test procedures

7.1 General

The equipment shall meet the requirements of ETSI EN 302 842-2 [6], clause 7.1.

7.2 Required test rig

An overview of the PCOs identified as required for the conduct of test cases is illustrated in figure 7.1.

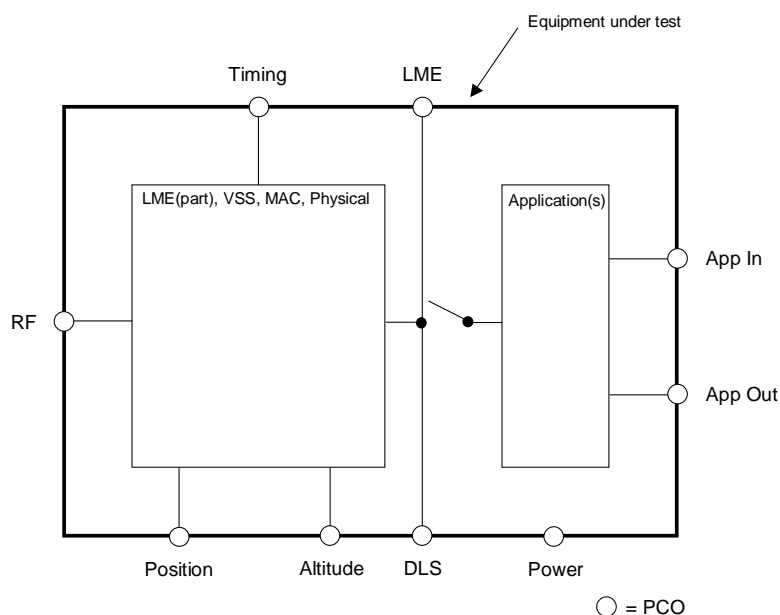


Figure 7.1: Location of PCOs

In addition, it shall be possible to verify that the equipment under test has passed the self test procedure.

The PCOs identified in this figure are the same as those described in ETSI EN 302 842-2 [6], clause 7.2 except as described below:

VSS:

- The VSS User PCO is not required and has therefore been removed.

DLS:

- The DLS PCO has been added.

LME:

- The LME PCO has been added.

7.3 Protocol test-suite description methodology

The equipment shall meet the requirements of ETSI EN 302 842-2 [6], clause 7.3.

7.4 Detailed protocol test procedures

The protocol test procedures shall be as described in ETSI EN 302 842-2 [6], clause 7.4 except as described below.

7.4.1 Test-suite overview

The test-suite overview shown in table 7.1 on the following pages lists the test cases by their name. The second column holds a short description of the test case objective. A cross reference between the test case names and the applicable requirements is provided in annex A.

Table 7.1: Protocol test-suite overview

| Test case name | Description |
|----------------------|--|
| DLS_Links | To demonstrate that a station supporting the communications functionality provided by the DLS will simultaneously support at least 8 peer-to-peer links with other stations. |
| DLS_ND2 | To demonstrate that a station will send data requiring a burst of length less than or equal to ND2 octets via the short transmission procedure, and a burst of length greater than ND2 via the long transmission procedure. |
| DLS_ND3 | To demonstrate that fragmentation will not occur if a message is less than or equal to ND3 but will occur if it is greater than ND3 in length. |
| DLS_Data | To demonstrate that a station will only transmit one data link packet in a DATA DLPDU. |
| DLS_Duplicate_Short | To demonstrate that duplicate DATA DLPDUs received by the short transmission procedure are discarded. |
| DLS_Duplicate_Long | To demonstrate that duplicate DATA DLPDUs received by the long transmission procedure are discarded. |
| DLS_Long_Order | To demonstrate that DATA DLPDUs which are part of the same fragmented packet are concatenated and delivered to the DLS user in the same order in which they appear in the packet. |
| DLS_Long_T_Receive_A | To demonstrate that a station correctly sets the T bit when receiving non-combined data packets from another station. |
| DLS_Long_T_Receive_B | To demonstrate that a station correctly sets the T bit when receiving combined data packets from another station. |
| DLS_Long_TM_Send_A | To demonstrate that a station correctly sets the T and M bits when sending a single data packet to another mobile station with the long transmission procedure. |
| DLS_Long_TM_Send_B | To demonstrate that a station correctly sets the T and M bits when sending multiple data packets in sequence to another station with the long transmission procedure. |
| DLS_Length | To demonstrate that the length subfield (lg) of an RTS correctly indicates the length in slots of the DLS burst containing a DATA DLPDU. |
| DLS_Priority_Long | To demonstrate that a sending station will maintain a prioritized queue of data packets for transmission, and will transmit the highest priority packet first, when data requiring the long transmission procedure is queued to be sent. |

| Test case name | Description |
|-----------------------------|---|
| DLS_Links | To demonstrate that a station supporting the communications functionality provided by the DLS will simultaneously support at least 8 peer-to-peer links with other stations. |
| DLS_Priority_Long_Short | To demonstrate that a sending station will maintain a prioritized queue of data packets for transmission, and will transmit the highest priority packet first, when data requiring a mixture of the long and short transmission procedures are queued to be sent. |
| DLS_Priority_Short | To demonstrate that a sending station will maintain a prioritized queue of data packets for transmission, and will transmit the highest priority packet first, when data requiring the short transmission procedure is queued to be sent. |
| DLS_Priority_CTRL | To demonstrate that a sending station will classify CTRL and CTRL_RTS as network management messages and assign the highest priority. |
| DLS_Priority_INFO_RTS | To demonstrate that an INFO_RTS will be assigned the same priority as the DATA DLPDU with which it is associated. |
| DLS_Long_UDATA_Send | To demonstrate that a single UDATA packet is sent correctly using the long transmission procedure, with the UDATA_RTS having the same priority and length fields as the DATA DLPDU with which it is associated. |
| DLS_Initialize_Send_Long | To demonstrate that a station displays correct behaviour and correctly sets the T and IB bits when sending data packets to another station during link initialization with the long transmission procedure. |
| DLS_INFO_RTS_SZOM_Send | To demonstrate that a station displays correct behaviour and correctly sets the T bit when sending data packets to another station during ZOCOP link initialization with the long transmission procedure. |
| DLS_INFO_SZOM_Send | To demonstrate that a station displays correct behaviour and correctly sets the T bit when sending data packets to another station during ZOCOP link initialization with the short transmission procedure. |
| DLS_INFO_RTS_SZOM_Receive | To demonstrate that a station displays correct behaviour and correctly sets the T bit when receiving data packets from another station during ZOCOP link initialization with the long transmission procedure. |
| DLS_INFO_SZOM_Receive | To demonstrate that a station displays correct behaviour and correctly sets the T bit when receiving data packets from another station during ZOCOP link initialization with the short transmission procedure. |
| DLS_TD1_Link | To demonstrate that a station that has established a ZOCOP link with a peer, and which has not sent a packet to that peer for TD1 seconds, and that wishes to send a DATA packet to the peer, will re-establish the link. |
| DLS_TD1_TD2_Link | To demonstrate that a station will regard the link as terminated if it has not transmitted to the same peer for TD1 seconds and if it has not received a transmission from the same peer for TD2 seconds. |
| DLS_Link_Terminated | To demonstrate that a station will regard the link as terminated if it receives a DM/DISC DLPDU or a DM/FRMR from the peer station. |
| DLS_SZOM_Retrans | To demonstrate that a station which transmitted an SZOM to a peer station will retransmit the SZOM in response to any DLPDU other than an SZOM, DM/DISC, DM/FRMR, or general failure, until it receives an SZOM. |
| DLS_ZOCOP_Link_CTRL | To demonstrate that if a CTRL_RTS (IB = 0), CTRL_CTS, CTRL, or CTRL_ACK DLPDU is received on an established ZOCOP link, then the receiver shall respond with a DM/FRMR in the reserved slot. |
| DLS_Initialize_Receive_Long | To demonstrate that a station displays correct behaviour and correctly sets the T bit when receiving data packets from another station during link initialization with the long transmission procedure. |
| DLS_Short_T_Receive | To demonstrate that a station will respond with an ACK with the T bit set correctly when it receives a DATA DLPDU via the short transmission procedure. |
| DLS_Short_T_Send | To demonstrate that a station correctly sets the T and M bits when sending a single data packet to another mobile station with the short transmission procedure. |
| DLS_Short_DATA_ACK | To demonstrate that a station which has data to send back to the sending station that would fit into a single slot with the ACK, will include its own data DLPDU with the ACK. |
| DLS_Long_DATA_ACK | To demonstrate that a station which has data that would not fit into a single slot with the ACK, will send back to the sending station an RTS DLPDU with the ACK. |
| DLS_Short_NoAck | To demonstrate that a station will retransmit the DLS burst, according to the default DLS retransmission parameters, if an acknowledgement is not received. |

| Test case name | Description |
|-----------------------------|---|
| DLS_Links | To demonstrate that a station supporting the communications functionality provided by the DLS will simultaneously support at least 8 peer-to-peer links with other stations. |
| DLS_Long_UDATA_Receive | To demonstrate that a station will correctly issue a CTS for an RTS_UDATA received via the long transmission procedure from another station. |
| DLS_Long_T_NoAck_A | To demonstrate that a station sends an ACK and correctly sets the T bit when in receipt of an RTS for a data packet that has already been received. |
| DLS_Long_Busy_A | To demonstrate that a station sends either a general confirm with a unicast reservation, or a general failure, when in receipt of an RTS at a time that the channel is too busy to transmit the requested data. |
| DLS_Long_NoInfo_A | To demonstrate that a station will not transmit in response to a CTS when it has no information to transmit. |
| DLS_Long_NACK_A | To demonstrate that a station sends a NACK when an expected data DLPDU is not received. |
| DLS_Long_NoLink | To demonstrate that a station in receipt of a CTRL_RTS with IB = 0 transmits a DM/FRMR when it does not have a link with the sender. |
| DLS_Short_NoLink | To demonstrate that a station in receipt of a CTRL transmits a DM/DISC when it does not have a link with the sender. |
| DLS_Long_IB_Error | To demonstrate that a station in receipt of a CTRL_RTS with IB = 1 and T = 1 transmits a DM/FRMR. |
| DLS_Long_SZOM_Error | To demonstrate that a station in receipt of an SZOM from a station with which it will only communicate using NSCOP will transmit a DM/FRMR. |
| DLS_Short_SZOM_Error | To demonstrate that a station in receipt of an SZOM from a station with which it will only communicate using NSCOP will transmit a DM/FRMR. |
| DLS_ND1_Short_Receive | To demonstrate that a station in receipt of a data packet from another station that is greater in length than ND1 will discard the packet. |
| DLS_ND1_Short_Send | To demonstrate that a station requested to send a data packet by a DLS user that is greater in length than ND1 will discard the packet. |
| LME_LE_Station_Choose | To demonstrate that a mobile station will choose an appropriate ground station to establish a link, and will then attempt to establish a link with that ground station by sending a CTRL_CMD_LE (re=1). |
| LME_LE_Ground_Invalid | To demonstrate that if the parameters in the CTRL_RSP_LE from the ground LME are not acceptable to the mobile LME, then the mobile LME shall transmit a DM/DISC to the ground. |
| LME_LE_Autotune | To demonstrate that if the Autotune parameter is included in the CTRL_RSP_LE and the mobile LME is unable to perform the autotune, then the mobile LME shall respond with an CTRL_CMD_LCR (re = 0). |
| LME_HO_Mob_Init | To demonstrate that a mobile that wishes to initiate a handoff to a new ground station will send to that ground station a CTRL_CMD_HO (re = 1) with parameters correctly defined. |
| LME_HO_Mob_Init_Auto | To demonstrate that in the case of a mobile-initiated handoff in which the ground station has included the autotune parameter in the CTRL_RSP_HO, and where the mobile cannot support the autotune, the mobile will transmit a CTRL_CMD_LCR (re = 0). |
| LME_HO_Mob_Init_Grd_Invalid | To demonstrate that in the case of a mobile-initiated handoff in which the parameters of the CTRL_RSP_HO transmitted by the ground station are not acceptable to the mobile, the mobile will transmit a DM/DISC to the ground on the new link. |
| LME_HO_TL2 | To demonstrate that a mobile LME will establish a link to a new ground station if TL2 seconds have elapsed since the LME initiated the request to send any burst to the current ground station. |
| LME_HO_TM2 | To demonstrate that a mobile LME will autonomously tune to an alternate frequency (provided in a frequency support list) and then establish a link to a new ground station, if timer TM2 expires. |
| LME_HO_Mob_Req_Grd_Init_A | To demonstrate that a mobile that wishes to request the ground LME to initiate a handoff, addressed to its current ground station, will send a CTRL_CMD_HO (re = 0) to that station. |
| LME_HO_Mob_Req_Grd_Init_B | To demonstrate that a mobile that wishes to request the ground LME to initiate a handoff, addressed to its proposed ground station, will send a CTRL_CMD_HO (re = 0) to that station. |
| LME_HO_Mob_Req_Grd_Init_Aff | To demonstrate that in the case of a mobile-requested ground-initiated handoff, the current link remains unaffected after transmission of the CTRL_CMD_HO (re = 0) while the mobile LME has not yet received a CTRL_CMD_HO (re = 1). |
| LME_HO_Gr_Req_Mob_Init | To demonstrate that a mobile that receives a CTRL_CMD_HO (re = 0) from a ground station will perform a mobile initiated handoff. |

| Test case name | Description |
|-------------------------------------|--|
| DLS_Links | To demonstrate that a station supporting the communications functionality provided by the DLS will simultaneously support at least 8 peer-to-peer links with other stations. |
| LME_HO_Gr_Req_Mob_Init_Invalid | To demonstrate that a mobile that receives an invalid CTRL_CMD_HO (re = 0) from a ground station will send a CTRL_CMD_LCR (re = 0). |
| LME_HO_Gr_Req_Mob_Init_Auto | To demonstrate that a mobile that receives a CTRL_CMD_HO (re = 0) from a ground station that includes the autotune parameter will perform a mobile initiated handoff on the new frequency. |
| LME_HO_Gr_Req_Mob_Init_Auto_Retrans | To demonstrate that a mobile that receives a CTRL_CMD_HO (re = 0) from a ground station that includes the autotune parameter will attempt to retransmit a CTRL_CMD_HO (re = 1) according to the retransmission procedures on the new frequency if it receives no response. |
| LME_HO_Receive_Invalid | To demonstrate that a mobile in receipt of a handoff command from a ground station with which it does not have a link will transmit a CTRL_LCR. |
| LME_HO_Ground_Init | To demonstrate that a station displays correct operation when handoff is initiated by the ground station. |
| LME_HO_Ground_Init_Invalid | To demonstrate that in the case of a ground-initiated handoff, a mobile LME which cannot accept the handoff request will transmit a CTRL_RSP_LCR. |
| LME_HO_Ground_Init_List | To demonstrate that a station displays correct operation when handoff is initiated by the ground station, and when the replacement ground station list parameter is provided by the ground station, and when a preferable ground station is included in the list. |
| LME_HO_Ground_Broadcast | To demonstrate that a station displays correct operation when requested to comply with a broadcast handoff. |
| LME_HO_GND_Broadcast_Invalid_A | To demonstrate that a mobile requested to comply with a ground requested broadcast handoff, where the mobile does not support broadcast handoff, will initiate either an air-initiated link handoff or request a link handoff. |
| LME_HO_GND_Broadcast_Invalid_B | To demonstrate that a mobile requested to comply with a ground requested broadcast handoff, where the ground station is not acceptable to the mobile, will initiate either an air-initiated link handoff or request a link handoff with a new ground station. |
| LME_HO_GND_Broadcast_Invalid_C | To demonstrate that a mobile requested to comply with a ground requested broadcast handoff, where the ground station address filter parameter does not equal the DLS address of a link that the mobile LME has, will not process the broadcast handoff. |
| LME_HO_GND_Broadcast_Invalid_D | To demonstrate that a mobile requested to comply with a ground requested broadcast handoff, where no mobile identifier subfield in a broadcast connection parameter equals its mobile address, will not process the broadcast handoff. |
| LME_TL1_Ground_Init | To demonstrate that a station displays correct operation of timer TL1 when handoff is initiated by the ground LME. |
| LME_TL1_Mob_Init | To demonstrate that a station displays correct operation of timer TL1 when handoff is initiated by the mobile. |
| LME_L1_Mob | To demonstrate that a station displays correct operation of counter L1. |

7.4.2 Declarations

For the performance of the tests, stimuli are applied and test results are observed at the Points of Control and Observation (PCO) as defined in clause 7.2.

7.4.3 Constraints

7.4.3.1 Abbreviations

7.4.3.1.1 Subfield mnemonics

Table 7.2: Subfield mnemonics

| Mnemonic | Meaning |
|----------|--------------------------|
| a | Additional slots |
| a/d | Autonomous/directed flag |
| ao | Acknowledgement offset |
| b | Broadcast link handoff |
| bd | Backoff delay |

| Mnemonic | Meaning |
|----------|----------------------------|
| c | CRC |
| cprf | CPR format even/odd |
| c/r | Command/response |
| d | Destination address |
| erid | Extended reservation ID |
| err | Error type |
| f | Frequency |
| flag | Flag delimiting burst |
| g | Ground station address |
| h | Link currently established |
| i | Supports initiated handoff |
| IB | Initialize Bit |
| id | Information field identity |
| in | Information field |
| lg | Length |
| M | More bit |
| mi | Message ID |
| neg | Negotiation subfield |
| ok | Confirm/failure flag |
| pr | Priority |
| r | Link connection refused |
| r-mi | Requested message ID |
| re | Response expected |
| res | Reserved bit |
| rid | Reservation ID |
| ro | Response offset |
| rrr | CTRL Retransmission number |
| s | Source address |
| sdf | Source/destination flag |
| seq | Sequence subfield |
| sss | CTRL Sequence number |
| T | Toggle bit |
| ver | Version number |

7.4.3.1.2 Special characters used in the subfield definitions

Table 7.3: Special characters used in the subfield definitions

| Character | Meaning |
|-----------|---|
| - | Subfield not applicable (0 bit length) |
| x | the value of this subfield is do not care |
| | The subfield is defined in an extra table |

7.4.3.1.3 Station addresses and positions

Station addresses are referred to in the test cases in the following format:

- add_A = address of station under test (station A);
- add_B = address of simulated station B;
- add_D = address of simulated station D;
- add_E = address of simulated station E;
- add_G = address of simulated (ground) station G;
- add_H = address of simulated (ground) station H.

The test station (station A) and other simulated stations are assumed to be at 0° latitude and at 0° longitude, unless otherwise specified. The positions of other stations are given in terms of the direction (East, E, is used for all cases) and distance in nautical miles (NM) with respect to the position of the station under test.

The test station (station A) and other simulated stations are assumed to be at an altitude of zero feet, unless otherwise specified.

The position of the stations and default Q2 parameters is as described in ETSI EN 302 842-2 [6], clause 7.4.3.1.3.

7.4.3.1.4 VDL bursts

The following burst formats exclude the preamble (for burst synchronization) and do not include the effect of bit stuffing. On generation of a burst at the RF PCO by the test harness, a logical 0 shall be inserted following a consecutive sequence of five logical 1s, except when arising in a flag. During recording of a burst by the test harness at the RF PCO, a sequence of five logical 1s followed by a 0 shall cause the 0 to be removed.

On generation of a burst, the test harness shall insert the value of the CRC field in accordance with clause 5.2.1.2.

Bursts are used in some test cases that are defined in ETSI EN 302 842-2 [6]. This clause defines additional bursts and frames not already defined in ETSI EN 302 842-2 [6].

Table 7.4: CTRL_RTS_a (CRa): CTRL_RTS DLPDU with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| lg, T, IB | 6 | 0 | T | IB | 0 | 1 | 0 | 0 | 1 |
| pr | 7 | pr ₄ | pr ₃ | pr ₂ | pr ₁ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| d | 8 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 9 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 10 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 11 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | 12 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 13 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | 14 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 13 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 14 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.5: INFO_RTS_a (IRa): INFO_RTS DLPDU with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| lg, T, IB | 6 | 0 | T | 0 | 1 | 1 | 0 | 0 | 1 |
| pr | 7 | pr ₄ | pr ₃ | pr ₂ | pr ₁ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| d | 8 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 9 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 10 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 11 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |

| Description | Octet | Bit number | | | | | | | |
|-------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ro | 12 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 13 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | 14 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 15 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 16 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.6: INFO_CTS_a (ICa): INFO_CTS DLPDU with information transfer reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| T | 6 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| ao | 7 | 0 | ao ₇ | ao ₆ | ao ₅ | ao ₄ | ao ₃ | ao ₂ | ao ₁ |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| ro | 9 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| ro, f | 10 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 11 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| d | 12 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 13 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 14 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 15 | 0 | 1 | 0 | 1 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 16 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 17 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.7: CTRL_CTS_a (CCa): CTRL_CTS DLPDU with information transfer reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| T, IB | 6 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| ao | 7 | 0 | ao ₇ | ao ₆ | ao ₅ | ao ₄ | ao ₃ | ao ₂ | ao ₁ |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| ro | 9 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| ro, f | 10 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 11 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| d | 12 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 13 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 14 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 15 | 0 | 1 | 0 | 1 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 16 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 17 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.8: UDATA_CTS_a (UDCa): UDATA_CTS DLPDU with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| lg, T, IB | 6 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| d | 7 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 8 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 9 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 10 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | 11 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 12 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | 13 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 14 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 15 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.9: INFO_a (IFa): INFO DLPDU with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| M, T, pr | 6 | M | T | pr ₄ | pr ₃ | pr ₂ | pr ₁ | 1 | 0 |
| in | 7 | information field | | | | | | | |
| d | n-8 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | n-7 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | n-6 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | n-5 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | n-4 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | n-3 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | n-2 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | n-1 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | n | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.10: CTRL_a (CTa): CTRL DLPDU with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| M, T, IB | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| in | 7 | information field | | | | | | | |
| d | n-8 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | n-7 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | n-6 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | n-5 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | n-4 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | n-3 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | n-2 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | n-1 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | n | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.11: CTRL_b (CTb): CTRL DLPDU with response reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-------------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| in | 7 | information field | | | | | | | |
| d | n-5 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | n-4 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | n-3 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | n-2 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | n-1 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | n | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.12: CTRL_ACK_a (CAa): CTRL_ACK DLPDU with response reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| T, IB | 6 | 0 | T | 0 | 0 | 0 | 0 | 0 | 1 |
| d | 8 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 9 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 10 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 11 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 12 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 13 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.13: INFO_b (IFb): INFO DLPDU with response reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-------------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| M, T, pr | 6 | M | T | pr ₄ | pr ₃ | pr ₂ | pr ₁ | 1 | 0 |
| in | 7 | information field | | | | | | | |
| d | n-5 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | n-4 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | n-3 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | n-2 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | n-1 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | n | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.14: INFO_ACK_a (IAa): INFO_ACK DLPDU with response reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| T | 6 | 0 | T | 0 | 1 | 0 | 0 | 0 | 1 |
| d | 7 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 8 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 9 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 10 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 11 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 12 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.15: INFO_RTS_SZOM_a (IRSza): INFO_RTS DLPDU and SZOM DLPDU combined burst with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| lg, T, IB | 6 | 0 | T | 0 | 1 | 1 | 0 | 0 | 1 |
| pr | 7 | pr ₄ | pr ₃ | pr ₂ | pr ₁ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| neg | 8 | neg ₄ | neg ₃ | neg ₂ | neg ₁ | 1 | 1 | 0 | 1 |
| seq | 9 | seq ₈ | seq ₇ | seq ₆ | seq ₅ | seq ₄ | seq ₃ | seq ₂ | seq ₁ |
| d | 10 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |

| Description | Octet | Bit number | | | | | | | |
|-------------|-------|------------------|------------------|------------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| d | 11 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 12 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 13 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | 14 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 15 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | 16 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 17 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 18 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

NOTE: In this table, octet 6 is the INFO_RTS and octets 8 and 9 are the SZOM DLPDU.

Table 7.16: INFO_CTS_SZOM_a (ICSZa): INFO_CTS DLPDU and SZOM DLPDU combined burst with response reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| T | 6 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| neg | 7 | neg ₄ | neg ₃ | neg ₂ | neg ₁ | 1 | 1 | 0 | 1 |
| seq | 8 | seq ₈ | seq ₇ | seq ₆ | seq ₅ | seq ₄ | seq ₃ | seq ₂ | seq ₁ |
| ao | 9 | 0 | ao ₇ | ao ₆ | ao ₅ | ao ₄ | ao ₃ | ao ₂ | ao ₁ |
| lg | 10 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| ro | 11 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| ro, f | 12 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 13 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| d | 14 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 15 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 16 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 17 | 0 | 1 | 0 | 1 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 18 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 19 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

NOTE: In this table, octet 6 is the INFO_CTS and octets 7 and 8 are the SZOM DLPDU.

Table 7.17: INFO_ACK_SZOM_a (IASZa): INFO_ACK DLPDU and SZOM DLPDU combined burst with response reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| T | 6 | 0 | T | 0 | 1 | 0 | 0 | 0 | 1 |
| neg | 7 | neg ₄ | neg ₃ | neg ₂ | neg ₁ | 1 | 1 | 0 | 1 |
| seq | 8 | seq ₈ | seq ₇ | seq ₆ | seq ₅ | seq ₄ | seq ₃ | seq ₂ | seq ₁ |
| d | 9 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 10 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 11 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 12 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 13 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 14 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

NOTE: In this table, octet 6 is the INFO_ACK and octets 7 and 8 are the SZOM DLPDU.

Table 7.18: INFO_SZOM_a (IFSZA): INFO DLPDU and SZOM DLPDU combined burst with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | pr ₄ | pr ₃ | pr ₂ | pr ₁ | 1 | 0 |
| in | 7 | information field | | | | | | | |
| neg | n-10 | neg ₄ | neg ₃ | neg ₂ | neg ₁ | 1 | 1 | 0 | 1 |
| seq | n-9 | seq ₈ | seq ₇ | seq ₆ | seq ₅ | seq ₄ | seq ₃ | seq ₂ | seq ₁ |
| d | n-8 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | n-7 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | n-6 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | n-5 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | n-4 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | n-3 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | n-2 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | n-1 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | n | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

NOTE: In this table, octets 6 and 7 are the INFO and octets n-10 and n-9 are the SZOM DLPDU.

Table 7.19: INFO_RTS_INFO_b (IRIFb): INFO_RTS and INFO DLPDU combined burst with response reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-------------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| lg, T, IB | 6 | 0 | T | 0 | 1 | 1 | 0 | 0 | 1 |
| pr | 7 | pr ₄ | pr ₃ | pr ₂ | pr ₁ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| M, T, pr | 8 | M | T | pr ₄ | pr ₃ | pr ₂ | pr ₁ | 1 | 0 |
| in | 9 | information field | | | | | | | |
| d | n-5 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | n-4 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | n-3 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | n-2 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | n-1 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | n | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

NOTE: In this table, octets 6 and 7 are the INFO_RTS and octets 8 and 9 are the INFO DLPDU.

Table 7.20: INFO_ACK_INFO_CTS_a (IAICa): INFO_ACK DLPDU and INFO_CTS DLPDU combined burst with information transfer reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| T | 6 | 0 | T | 0 | 1 | 0 | 0 | 0 | 1 |
| T | 7 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| ao | 8 | 0 | ao ₇ | ao ₆ | ao ₅ | ao ₄ | ao ₃ | ao ₂ | ao ₁ |
| lg | 9 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| ro | 10 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| ro, f | 11 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 12 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| d | 13 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 14 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 15 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 16 | 0 | 1 | 0 | 1 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 17 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 18 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

NOTE: In this table, octet 6 is the INFO_ACK and octet 7 is the INFO_CTS DLPDU.

Table 7.21: DM_DISC_a (DMDCa): DM/DISC DLPDU with response reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| burst id | 6 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| d | 7 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 8 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 9 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 10 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 11 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 12 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.22: UINFO_a (UIa): UINFO DLPDU with response reservation
with address type field equal to 7

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | ud1 ₅ | ud1 ₄ | ud1 ₃ | ud1 ₂ | ud1 ₁ | 1 | 1 | 1 |
| in | 7 | information field | | | | | | | |
| erid, d | n-2 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | n-1 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | n | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.23: UCTRL_a (UCa): UCTRL DLPDU with response reservation
with address type field equal to 7

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | ucid ₅ | ucid ₄ | ucid ₃ | ucid ₂ | ucid ₁ | 0 | 1 | 1 |
| in | 7 | information field | | | | | | | |
| erid, d | n-2 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | n-1 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | n | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.24: UCTRL_b (UCb): UCTRL DLPDU with connection management, control sequencing, ground station address filter, broadcast connection, ATN router nets and response reservation with address type field equal to 7

| Description | Octet | Bit number | | | | | | | |
|------------------------|-------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | ucid ₅ | ucid ₄ | ucid ₃ | ucid ₂ | ucid ₁ | 0 | 1 | 1 |
| Cxn management | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 7 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 8 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 9 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 11 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| Protocol options | 12 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| length | 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| b, i | 14 | 0 | 0 | 0 | 0 | 0 | b | i | 0 |
| Grd stn address filter | 15 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| length | 16 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| g | 17 | 0 | 0 | 0 | 0 | 0 | g ₂₇ | g ₂₆ | g ₂₅ |
| g | 18 | g ₂₄ | g ₂₃ | g ₂₂ | g ₂₁ | g ₂₀ | g ₁₉ | g ₁₈ | g ₁₇ |
| g | 19 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 20 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| Broadcast connection | 21 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| lg | 22 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| a | 23 | 0 | 0 | 0 | 0 | 0 | a ₂₇ | a ₂₆ | a ₂₅ |
| a | 24 | a ₂₄ | a ₂₃ | a ₂₂ | a ₂₁ | a ₂₀ | a ₁₉ | a ₁₈ | a ₁₇ |
| a | 25 | a ₁₆ | a ₁₅ | a ₁₄ | a ₁₃ | a ₁₂ | a ₁₁ | a ₁₀ | a ₉ |
| a | 26 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ |
| ATN router NETs | 27 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 28 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| a | 29 | a ₂₄ | a ₂₃ | a ₂₂ | a ₂₁ | a ₂₀ | a ₁₉ | a ₁₈ | a ₁₇ |
| a | 30 | a ₁₆ | a ₁₅ | a ₁₄ | a ₁₃ | a ₁₂ | a ₁₁ | a ₁₀ | a ₉ |
| a | 31 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ |
| r | 32 | r ₂₄ | r ₂₃ | r ₂₂ | r ₂₁ | r ₂₀ | r ₁₉ | r ₁₈ | r ₁₇ |
| r | 33 | r ₁₆ | r ₁₅ | r ₁₄ | r ₁₃ | r ₁₂ | r ₁₁ | r ₁₀ | r ₉ |
| r | 34 | r ₈ | r ₇ | r ₆ | r ₅ | r ₄ | r ₃ | r ₂ | r ₁ |
| erid, d | 35 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 36 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 37 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.25: GEN_RESP_b (GRb): General response burst with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|------------------|-------|------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| s, ver, rid, a/d | 1 | s ₂₇ | s ₂₆ | s ₂₅ | 0 | 0 | 0 | 0 | 1 |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| ok, mi | 5 | ok | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| res, r-mi | 6 | 0 | r-mi ₇ | r-mi ₆ | r-mi ₅ | r-mi ₄ | r-mi ₃ | r-mi ₂ | r-mi ₁ |
| res | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| bd | 8 | bd ₈ | bd ₇ | bd ₆ | bd ₅ | bd ₄ | bd ₃ | bd ₂ | bd ₁ |
| err | 9 | err ₈ | err ₇ | err ₆ | err ₅ | err ₄ | err ₃ | err ₂ | err ₁ |
| d | 10 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 11 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 12 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 13 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | 14 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 15 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | 16 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 17 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 18 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.26: INFO_ACK_INFO_a (IAIFa): INFO_ACK and INFO DLPDU combined burst with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| T | 6 | 0 | T | 0 | 1 | 0 | 0 | 0 | 1 |
| pr | 7 | M | T | pr ₄ | pr ₃ | pr ₂ | pr ₁ | 1 | 0 |
| in | 8 | information field | | | | | | | |
| d | n-8 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | n-7 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | n-6 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | n-5 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | n-4 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | n-3 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | n-2 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | n-1 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | n | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

NOTE: In this table, octet 6 is the INFO_ACK and octets 7 and 8 are the INFO DLPDU.

Table 7.27: INFO_ACK_INFO_RTS_a (IAIRa): INFO_ACK and INFO_RTS DLPDU combined burst with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| T | 6 | 0 | T | 0 | 1 | 0 | 0 | 0 | 1 |
| lg, T | 7 | 0 | T | 0 | 1 | 1 | 0 | 0 | 1 |
| pr, lg | 8 | pr ₄ | pr ₃ | pr ₂ | pr ₁ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| d | 9 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 10 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 11 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 12 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | 13 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 14 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | 15 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 16 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 17 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

NOTE: In this table, octet 6 is the INFO_ACK and octets 7 and 8 are the INFO_RTS DLPDU.

Table 7.28: CTRL_c (CTc): CTRL DLPDU with connection management, control sequencing, and unicast reservation

| Description | Octet | Bit number | | | | | | | |
|-----------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| M, T, IB | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| d | 13 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 14 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 15 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 16 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | 17 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 18 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | 19 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 20 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 21 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.29: CTRL_d (CTd): CTRL DLPDU with connection management, control sequencing, and response reservation

| Description | Octet | Bit number | | | | | | | |
|-----------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| d | 13 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 14 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 15 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 16 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 17 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 18 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.30: CTRL_e (CTe): Invalid CTRL DLPDU with connection management, control sequencing, and response reservation

| Description | Octet | Bit number | | | | | | | |
|-----------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 1 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| d | 13 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 14 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 15 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 16 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 17 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 18 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.31: CTRL_g (CTg): Invalid CTRL DLPDU with connection management, control sequencing, and unicast reservation

| Description | Octet | Bit number | | | | | | | |
|-----------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| M, T, IB | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 1 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| d | 13 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 14 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 15 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 16 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | 17 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 18 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | 19 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 20 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 21 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.32: CTRL_h (CTh): CTRL DLPDU with connection management, control sequencing, ATN router NETs, ground-based system mask, and response reservation

| Description | Octet | Bit number | | | | | | | |
|--------------------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| Protocol options | 13 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| length | 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| b, i | 15 | 0 | 0 | 0 | 0 | 0 | b | i | 0 |
| ATN router NETs | 16 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 17 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| a | 18 | a ₂₄ | a ₂₃ | a ₂₂ | a ₂₁ | a ₂₀ | a ₁₉ | a ₁₈ | a ₁₇ |
| a | 19 | a ₁₆ | a ₁₅ | a ₁₄ | a ₁₃ | a ₁₂ | a ₁₁ | a ₁₀ | a ₉ |
| a | 20 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ |
| r | 21 | r ₂₄ | r ₂₃ | r ₂₂ | r ₂₁ | r ₂₀ | r ₁₉ | r ₁₈ | r ₁₇ |
| r | 22 | r ₁₆ | r ₁₅ | r ₁₄ | r ₁₃ | r ₁₂ | r ₁₁ | r ₁₀ | r ₉ |
| r | 23 | r ₈ | r ₇ | r ₆ | r ₅ | r ₄ | r ₃ | r ₂ | r ₁ |
| Ground-based system mask | 24 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| lg | 25 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| m | 26 | 0 | 0 | 0 | 0 | 0 | g ₂₅ | g ₂₆ | g ₂₇ |
| g | 27 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| g | 28 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 29 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| d | 30 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 31 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 32 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 33 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 34 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 35 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.33: CTRL_i (CTi): CTRL DLPDU with connection management, control sequencing, ATN router NETs, ground-based system mask, replacement ground station list, and response reservation

| Description | Octet | Bit number | | | | | | | |
|---------------------------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| ATN router NETs | 13 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 14 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| a | 15 | a ₂₄ | a ₂₃ | a ₂₂ | a ₂₁ | a ₂₀ | a ₁₉ | a ₁₈ | a ₁₇ |
| a | 16 | a ₁₆ | a ₁₅ | a ₁₄ | a ₁₃ | a ₁₂ | a ₁₁ | a ₁₀ | a ₉ |
| a | 17 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ |
| r | 18 | r ₂₄ | r ₂₃ | r ₂₂ | r ₂₁ | r ₂₀ | r ₁₉ | r ₁₈ | r ₁₇ |
| r | 19 | r ₁₆ | r ₁₅ | r ₁₄ | r ₁₃ | r ₁₂ | r ₁₁ | r ₁₀ | r ₉ |
| r | 20 | r ₈ | r ₇ | r ₆ | r ₅ | r ₄ | r ₃ | r ₂ | r ₁ |
| Ground-based system mask | 21 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| lg | 22 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| m | 23 | 0 | 0 | 0 | 0 | 0 | g ₂₅ | g ₂₆ | g ₂₇ |
| g | 24 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| g | 25 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 26 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| Replacement ground station list | 27 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| lg | 28 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| g | 29 | 0 | 0 | 0 | 0 | 0 | g ₂₇ | g ₂₆ | g ₂₅ |
| g | 30 | g ₂₄ | g ₂₃ | g ₂₂ | g ₂₁ | g ₂₀ | g ₁₉ | g ₁₈ | g ₁₇ |
| g | 31 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 32 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| d | 33 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 34 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 35 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 36 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 37 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 38 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.34: CTRL_k (CTk): CTRL DLPDU with connection management, control sequencing, autotune, replacement ground station list, and response reservation

| Description | Octet | Bit number | | | | | | | |
|-----------------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| Autotune | 13 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| length | 14 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| m, f | 15 | m ₄ | m ₃ | m ₂ | m ₁ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 16 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| Repl Grd Station List | 17 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| lg | 18 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| g | 19 | 0 | 0 | 0 | 0 | 0 | g ₂₇ | g ₂₆ | g ₂₅ |
| g | 20 | g ₂₄ | g ₂₃ | g ₂₂ | g ₂₁ | g ₂₀ | g ₁₉ | g ₁₈ | g ₁₇ |
| g | 21 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 22 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| d | 23 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 24 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 25 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 26 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 27 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 28 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.35: CTRL_ACK_CTRL_RTS_a (CACRa): CTRL_ACK and CTRL_RTS DLPDU combined burst with unicast reservation

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| T | 6 | 0 | T | 0 | 0 | 0 | 0 | 0 | 1 |
| lg, T, IB | 7 | 0 | T | IB | 0 | 1 | 0 | 0 | 1 |
| pr, lg | 8 | 0 | 0 | 0 | 0 | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| d | 9 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 10 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 11 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 12 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | 13 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 14 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | 15 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 16 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 17 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

NOTE: In this table, octet 6 is the CTRL_ACK and octets 7 and 8 are the CTRL_RTS DLPDU.

Table 7.36: CTRL_I (CTI): CTRL DLPDU with connection management, control sequencing, replacement ground station list, and response reservation

| Description | Octet | Bit number | | | | | | | |
|-----------------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| Repl Grd Station List | 13 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| lg | 14 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| g | 15 | 0 | 0 | 0 | 0 | 0 | g ₂₇ | g ₂₆ | g ₂₅ |
| g | 16 | g ₂₄ | g ₂₃ | g ₂₂ | g ₂₁ | g ₂₀ | g ₁₉ | g ₁₈ | g ₁₇ |
| g | 17 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 18 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| d | 19 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 20 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 21 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 22 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 23 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 24 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.37: CTRL_m (CTm): CTRL DLPDU with connection management, control sequencing, ground-based system mask, and response reservation

| Description | Octet | Bit number | | | | | | | |
|--------------------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| M, T, IB | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| Ground-based system mask | 13 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| lg | 14 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| m | 15 | 0 | 0 | 0 | 0 | 0 | g ₂₅ | g ₂₆ | g ₂₇ |
| g | 16 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| g | 17 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 18 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| d | 19 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 20 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 21 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 22 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 23 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 24 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.38: CTRL_n (CTn): Invalid CTRL DLPDU with connection management, control sequencing, ground-based system mask and unicast reservation

| Description | Octet | Bit number | | | | | | | |
|--------------------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| M, T, IB | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 1 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| Ground-based system mask | 13 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| lg | 14 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| m | 15 | 0 | 0 | 0 | 0 | 0 | g ₂₅ | g ₂₆ | g ₂₇ |
| g | 16 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| g | 17 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 18 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| d | 19 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 20 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 21 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 22 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | 23 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 24 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| erid, pr | 25 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 26 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 27 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.39: CTRL_o (CTo): CTRL DLPDU with connection management, control sequencing, autotune, replacement ground station list, ground-based system mask and response reservation

| Description | Octet | Bit number | | | | | | | |
|--------------------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| Autotune | 13 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| length | 14 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| m, f | 15 | m ₄ | m ₃ | m ₂ | m ₁ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 16 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| Repl Grd Station List | 17 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| lg | 18 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| g | 19 | 0 | 0 | 0 | 0 | 0 | g ₂₇ | g ₂₆ | g ₂₅ |
| g | 20 | g ₂₄ | g ₂₃ | g ₂₂ | g ₂₁ | g ₂₀ | g ₁₉ | g ₁₈ | g ₁₇ |
| g | 21 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 22 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| Ground-based system mask | 23 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| lg | 24 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| m | 25 | 0 | 0 | 0 | 0 | 0 | g ₂₅ | g ₂₆ | g ₂₇ |
| g | 26 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| g | 27 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 28 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| d | 29 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 30 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 31 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 32 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 33 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 34 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.40: CTRL_q (CTq): CTRL DLPDU with connection management, control sequencing, protocol options and response reservation

| Description | Octet | Bit number | | | | | | | |
|------------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| Protocol options | 13 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| length | 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| b, i | 15 | 0 | 0 | 0 | 0 | 0 | b | i | 0 |
| d | 16 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 17 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 18 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 19 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 20 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 21 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.41: CTRL_r (CTr): Invalid CTRL DLPDU with connection management, control sequencing, ATN router NETs, ground-based system mask and response reservation

| Description | Octet | Bit number | | | | | | | |
|--------------------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 1 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| Protocol options | 13 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| length | 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| b, i | 15 | 0 | 0 | 0 | 0 | 0 | b | i | 0 |
| ATN router NETs | 16 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 17 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| a | 18 | a ₂₄ | a ₂₃ | a ₂₂ | a ₂₁ | a ₂₀ | a ₁₉ | a ₁₈ | a ₁₇ |
| a | 19 | a ₁₆ | a ₁₅ | a ₁₄ | a ₁₃ | a ₁₂ | a ₁₁ | a ₁₀ | a ₉ |
| a | 20 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ |
| r | 21 | r ₂₄ | r ₂₃ | r ₂₂ | r ₂₁ | r ₂₀ | r ₁₉ | r ₁₈ | r ₁₇ |
| r | 22 | r ₁₆ | r ₁₅ | r ₁₄ | r ₁₃ | r ₁₂ | r ₁₁ | r ₁₀ | r ₉ |
| r | 23 | r ₈ | r ₇ | r ₆ | r ₅ | r ₄ | r ₃ | r ₂ | r ₁ |
| Ground-based system mask | 24 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| lg | 25 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| m | 26 | 0 | 0 | 0 | 0 | 0 | g ₂₅ | g ₂₆ | g ₂₇ |
| g | 27 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| g | 28 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 29 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| d | 30 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 31 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 32 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 33 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 34 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 35 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.42: CTRL_s (CTs): CTRL DLPDU with connection management, control sequencing, LCR cause, and response reservation

| Description | Octet | Bit number | | | | | | | |
|-----------------|-------|-----------------|-----------------|-----------------|------------------|------------------|------------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| pr | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| LCR cause | 13 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| lg | 14 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| c | 15 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| d | 16 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 17 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| a | 18 | a ₈ | a ₇ | a ₆ | a ₅ | a ₄ | a ₃ | a ₂ | a ₁ |
| d | 13 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 14 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 15 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 16 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 17 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 18 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.43: CTRL_t (CTt): CTRL DLPDU with connection management, control sequencing, ground-based system mask, random access and response reservation

| Description | Octet | Bit number | | | | | | | |
|--------------------------|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| a/d, rid, ver | 1 | s ₂₇ | s ₂₆ | s ₂₅ | ver ₃ | ver ₂ | ver ₁ | rid | a/d |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ |
| burst id | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| M, T, IB | 6 | M | T | re | c/r | 0 | 0 | 0 | 0 |
| Cxn management | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg | 8 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| r, h | 9 | 0 | 0 | 0 | 0 | 0 | 0 | r | h |
| CTRL sequencing | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| length | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| rrr, sss | 12 | r ₄ | r ₃ | r ₂ | r ₁ | s ₄ | s ₃ | s ₂ | s ₁ |
| Ground-based system mask | 13 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| lg | 14 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| m | 15 | 0 | 0 | 0 | 0 | 0 | g ₂₅ | g ₂₆ | g ₂₇ |
| g | 16 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| g | 17 | g ₁₆ | g ₁₅ | g ₁₄ | g ₁₃ | g ₁₂ | g ₁₁ | g ₁₀ | g ₉ |
| g | 18 | g ₈ | g ₇ | g ₆ | g ₅ | g ₄ | g ₃ | g ₂ | g ₁ |
| Random access | 19 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| length | 20 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| p | 21 | p ₈ | p ₇ | p ₆ | p ₅ | p ₄ | p ₃ | p ₂ | p ₁ |
| VS3 | 22 | VS3 ₈ | VS3 ₇ | VS3 ₆ | VS3 ₅ | VS3 ₄ | VS3 ₃ | VS3 ₂ | VS3 ₁ |
| VS3 | 23 | VS3 ₁₆ | VS3 ₁₅ | VS3 ₁₄ | VS3 ₁₃ | VS3 ₁₂ | VS3 ₁₁ | VS3 ₁₀ | VS3 ₉ |
| TM2 | 24 | TM2 ₈ | TM2 ₇ | TM2 ₆ | TM2 ₅ | TM2 ₄ | TM2 ₃ | TM2 ₂ | TM2 ₁ |
| d | 25 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 26 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 27 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf | 28 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 29 | c ₉ | c ₁₀ | c ₁₁ | c ₁₂ | c ₁₃ | c ₁₄ | c ₁₅ | c ₁₆ |
| c | 30 | c ₁ | c ₂ | c ₃ | c ₄ | c ₅ | c ₆ | c ₇ | c ₈ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

7.4.3.2 Test cases

The equipment under test shall be brought into the defined idle state before the performance of the individual test cases. Each test case starts in this state and leaves the equipment in that state after completion. The idle state is the state which the equipment enters after successful completion of the power up sequence. To perform several test cases in sequence the power on macro M_POWER_UP shall only be executed at the beginning.

All protocol test cases shall be performed on a GSC channel (GSC1 or GSC2) unless stated otherwise in the test case itself.

If an expected test result mentioned in a test step is not observed during the execution of a test case, then the test case shall be terminated and the equipment initialized before a new test case is executed. Further verification in that test case may not provide any valid results.

7.4.3.2.1 Test case macros

Test case macros shall be as defined in ETSI EN 302 842-2 [6], clause 7.4.3.2.1 with the following additional macros.

| Macro name: M_INITIALIZE_NSCOP | | | | Performs initial exchange for NSCOP communication. | | |
|--------------------------------|------|--------|-----|--|-----|---|
| Parameters: | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| macro | 1 | send | RF | CTRL_RTS_a (s:= add_B; d:= add_A; IB:= 1; T:= 0; lg:= 4) | CRa | Send an RTS using the long transmission procedures from a simulated station B, with IB = 1 and T = 0, indicating this is the first transmission from station B to station A. The RTS contains a unicast reservation for a response. |
| | 2 | verify | RF | CTRL_CTS_a (s = add_A; d = add_B) in the slot reserved by the RTS | CCa | Verify that a CTRL_CTS is sent by the station under test in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 3 | send | RF | CTRL_b (s:= add_B; d:= add_A; T:= 0) in the slot reserved by the CTS | CTb | Send a CTRL DLPDU from station B with T = 0 in the slot reserved by the CTS. |
| | 4 | verify | RF | CTRL_ACK_a (s = add_A; d = add_B; T = 0) in the ack slot reserved by the CTS | CAa | Verify that a CTRL_ACK is sent by the station under test in the ack slot reserved by the CTS. |
| Comments: | | | | | | |

| Macro name: M_INITIALIZE_ZOCOP | | | | Performs initial exchange for ZOCOP communication. | | |
|--------------------------------|------|--------|-----|--|-------|---|
| Parameters: | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| macro | 1 | send | RF | INFO_SZOM_a (s:= add_B; d:= add_A; T:= 0; neg:= 0; lg:= 4) | IFSZa | Send an INFO_SZOM using the short transmission procedures from a simulated station B, with T = 0, neg = 0 and lg = 4, indicating this is the first transmission from station B to station A. The INFO_SZOM contains a unicast reservation for a response. |
| | 2 | verify | RF | INFO_ACK_SZOM_a (s = add_A; d = add_B) in the slot reserved by the INFO_SZOM | IASZa | Verify that an INFO_ACK_SZOM is sent by the station under test in the slot reserved by the INFO_SZOM. |
| | 3 | verify | RF | T = 0, neg = 0 and seq = 0 in INFO_ACK_SZOM_a (s = add_A; d = add_B) | IASZa | Verify that the INFO_ACK_SZOM contains T = 0, neg = 0 and seq = 0. |
| Comments: | | | | | | |

| Macro name: M_LME_INITIALIZE_LINK_MOB | | | | Performs link establishment at the LME. | | |
|---------------------------------------|------|--------|-----|---|-------|--|
| Parameters: | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| macro | 1 | send | LME | REQUEST TO PERFORM LINK ESTABLISHMENT with ground station G | | Send a request at the LME of the station under test to perform link establishment with the selected simulated ground station G. |
| | 2 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 3 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_G; IB:= 1; T:= 0; lg:= 1) | CRA | Verify that the station under test sends an RTS using the long transmission procedures to simulated ground station G, with IB = 1 and T = 0, indicating this is the first transmission from station A to station G. The RTS contains a unicast reservation for a response. |
| | 4 | send | RF | CTRL_CTS_a (s = add_G; d = add_A; T = 0) in the slot reserved by the RTS | CCa | Send a CTS to the station under test from simulated ground station G. |
| | 5 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 6 | verify | RF | CTRL_d (s:= add_A; d:= add_G; M:= 0; T:= 0; re:= 1; c/r:= 0; h:= 0; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Verify that the station under test sends a CTRL DLPDU to simulated ground station G with parameters indicating a CTRL_CMD_LE (re = 1). |
| | 7 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_G; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS to the station under test. |
| | 8 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 9 | verify | RF | CTRL_CTS_a (s = add_A; d = add_G) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to simulated ground station G. |
| | 10 | send | RF | CTRL_m (s:= add_G; d:= add_A; M:= 0; T:= 0; re:= 1; c/r:= 1; h:= 0; r:= 0; lg:= 1; rrr:= 0; sss:= 0; g:= add_G) in the slot reserved by the CTS | CTm | Send a CTRL burst indicating a CTRL_RSP_LE to the station under test, including the ground-based system mask parameter. |
| | 11 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 12 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_G; T:= 0) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK in response to the completed message. |
| Comments: | | | | | | |

7.4.3.2.2 Test case descriptions

| Test case name: DLS_Links | | | | | | |
|---|------|---------|-----|--|-----|--|
| Purpose: To demonstrate that a station supporting the communications functionality provided by the DLS will simultaneously support at least 8 peer-to-peer links with other stations. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep | | s1:= {B, C, D, E, F, G, H, I} | | Start a loop for 8 simulated stations to send messages to station A. |
| | 3 | send | RF | CTRL_RTS_a (s:= s1; IB:= 1; T:= 0; d:= add_A) | CRa | Send a CTRL_RTS from station s1 to station A. The burst contains a unicast reservation reserving a slot for A's response. |
| | 4 | end rep | | next s1 | | Select the next station from which to send transmissions. |
| | 5 | rep | | s1:= {B, C, D, E, F, G, H, I} | | Start a loop to check messages sent in response by station A. |
| | 6 | verify | RF | CTRL_CTS_a (s = add_A; d = s1) in slot reserved by the RTS from station s1 | CCa | Verify that CTRL_CTS messages are sent in response by station A in the slot reserved by the appropriate RTS, and hence that 8 links are being supported. |
| | 7 | end rep | | next s1 | | End loop. |
| postamble | 8 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_ND2 | | | | |
|-----------------|------|---|-----|--|-----|---|
| Purpose: | | To demonstrate that a station will send data requiring a burst of length less than or equal to ND2 octets via the short transmission procedure, and a burst of length greater than ND2 via the long transmission procedure. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| test body | 4 | rep 3 | | I1:= {1, 2, 3} | | Start loop. |
| | 5 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B) with length requiring burst = I1 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU to a simulated station B requiring a burst of length I1 slots. |
| | 6 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 7 | verify | RF | INFO_a (s = add_A; d = add_B) | IFa | Verify that the station under test sends a DATA DLPDU by the short transmission procedures. The burst contains a unicast reservation for an acknowledgement. |
| | 8 | verify | RF | M = 0 in INFO_a | IFa | Verify that the more bit M = 0 in the DATA DLPDU transmitted by the short transmission procedure. |
| | 9 | send | RF | INFO_ACK (s:= add_B; d:= add_A) in slot reserved by the INFO | IAa | Send an ACK from station B in the slot reserved by the INFO to the station under test. |
| | 10 | endrep | | next I1 | | End loop. |
| | 11 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B) with length requiring burst = 4 slots | | Send a request from a DLS user to the station under test to transmit a DATA DLPDU to a simulated station B, requiring a burst of length 4 slots. |
| | 12 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 13 | verify | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Verify that an RTS is sent by the station under test, indicating initiation of a long transmission procedure. The RTS contains a unicast reservation for a response. |
| | 14 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 15 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 16 | verify | RF | INFO_b (s = add_A; d = add_B) in slot reserved by the CTS | IFb | Verify that a DATA DLPDU is sent by station A in the reserved slot. |
| | 17 | verify | RF | M = 0 in INFO_b | IFb | Verify that the more bit M = 0 in the DATA DLPDU transmitted by the long transmission procedure. |
| | 18 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the ack slot reserved by the CTS | IAa | Send an ACK from station B in response in the slot reserved for the acknowledgement. |
| postamble | 19 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_ND3 | | | | |
|-----------------|------|--|-----|---|-------|---|
| Purpose: | | To demonstrate that fragmentation will not occur if a message is less than or equal to ND3 but will occur if it is greater than ND3 in length. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 6 slots; ND2:= 86 octets) | | Set the parameter ND3 = 6 and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| Test body | 4 | rep 3 | | I1:= {4, 5, 6} | | Start loop. |
| | 5 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B) with length = I1 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU requiring a length of I1 slots. |
| | 6 | verify | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Verify that an RTS is sent by the station under test, indicating initiation of a long transmission procedure. The burst contains a unicast reservation reserving a slot for a response. |
| | 7 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) in slot reserved by RTS | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 8 | verify | RF | INFO_b (s = add_A; d = add_B) in slot reserved by CTS | IFb | Verify that a DATA DLPDU is sent by station A in the slot reserved by the CTS. |
| | 9 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) | IAa | Send an ACK from station B in the ack slot reserved by the CTS in response to the completed message. |
| | 10 | endrep | | next I1 | | End loop. |
| | 11 | rep 3 | | I1:= {7, 8, 9} | | Start loop. |
| | 12 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d := add_B) with length = I1 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU requiring a length of I1 slots. |
| | 13 | verify | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Verify that an RTS is sent by the station under test, indicating the beginning of a long transmission procedure. The RTS contains a unicast reservation for a response. |
| | 14 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 15 | verify | RF | INFO_RTS_INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IRIFb | Verify that an RTS / DATA is sent by station A in the slot reserved by the CTS. The RTS reserves another slot for the rest of the data. |
| | 16 | send | RF | INFO_ACK_INFO_CTS_a (s:= add_B; d:= add_A) in the ack slot reserved by the previous CTS | IAICa | Send an ACK / CTS from station B in response, in the ack slot reserved by the previous CTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 17 | verify | RF | INFO_b (s = add_A; d = add_B) in the slot reserved by the last CTS | IFb | Verify that the DATA DLPDU is sent by station A in the slot reserved by the last CTS. |
| | 18 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the ack slot reserved by the last CTS | IAa | Send an ACK from station B in response to the completed message, in the ack slot reserved by the last CTS. |
| | 19 | endrep | | next I1 | | End loop. |
| postamble | 20 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Data | | | | |
|-----------------|------|--|-----|---|-----|--|
| Purpose: | | To demonstrate that a station will only transmit one data link packet in a DATA DLPDU. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:=5 slots; ND2:= 4 slots) | | Set parameter ND3 = 5 and ND2 = 118 octets (4 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| test body | 4 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B; pr:= 2) with length = 2 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU with length = 2 slots. |
| | 5 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B; pr:= 3) with length = 2 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU with length = 2 slots. |
| | 6 | verify | RF | INFO_a (s = add_A; d = add_B; pr = 3)) | CTa | Verify that a DATA DLPDU is sent by the station under test, indicating the beginning of a short transmission procedure. The burst includes a unicast request reservation for a response. |
| | 7 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the slot reserved by the INFO with pr = 3 | CAa | Send an ACK from station B as an acknowledgement in the slot reserved by the INFO with pr = 3. |
| | 8 | verify | RF | INFO_a (s = add_A; d = add_B; pr = 2) | CTa | Verify that a DATA DLPDU is sent by station A indicating the beginning of a short transmission procedure. The burst includes a unicast request reservation for a response. |
| | 9 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the slot reserved by the INFO with pr = 2 | CAa | Send an ACK from station B as an acknowledgement in the slot reserved by the INFO with pr = 2. |
| postamble | 10 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | | DLS_Duplicate_Short | | | |
|-----------------|------|---|---------------------|---|-----|--|
| Purpose: | | To demonstrate that duplicate DATA DLPDUs received by the short transmission procedure are discarded. | | | | |
| Reference: | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| test body | 3 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 1; in:= info1) | IFa | Send a message with T = 1 and information field = info1 from a simulated station B to the station under test via the short transmission procedure. The burst contains a unicast reservation reserving a slot for a response. |
| | 4 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 1; in:= info2) | IFa | Send a second message with T = 1 and information field = info2 from station B to the station under test. The burst contains a unicast reservation reserving a slot for a response. |
| | 5 | verify | DLS | INFO_a (s = add_B; d = add_A; T:= 1; in = info1) passed to DLS user | IFa | Verify that only the first INFO_a message is passed to the DLS user by the station under test and therefore that the duplicate message was discarded. |
| | 6 | verify | RF | INFO_ACK (s = add_A; d = add_B) sent by the station under test in the slot reserved by the first INFO | IAa | Verify that an ACK is sent to station B by the station under test in the slot reserved by the first INFO. |
| | 7 | verify | RF | INFO_ACK (s = add_A; d = add_B) sent by the station under test in the slot reserved by the INFO message | IAa | Verify that an ACK is sent to station B by the station under test in the slot reserved by the second INFO. |
| postamble | 8 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Duplicate_Long | | | | |
|-----------------|------|--|-----|---|-----|--|
| Purpose: | | To demonstrate that duplicate DATA DLPDUs received by the long transmission procedure are discarded. | | | | |
| Reference: | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the air-ground link. |
| test body | 4 | send | RF | INFO_RTS_a (s:= add_B; d:= add_A; T:= 1; pr:= 3; lg:= 5) | IRa | Send an INFO_RTS using the long transmission procedures from a simulated station B, with T = 1, indicating this is the first transmission in the sequence from station B to station A following the initialization. The burst contains a unicast reservation for a response. |
| | 5 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 6 | verify | RF | INFO_CTS_a (s = add_A; d = add_B) in slot reserved by RTS | ICa | Verify that a CTS is transmitted by station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 7 | send | RF | INFO_b (s:= add_B; d:= add_A; T:= 1) in slot reserved by CTS | IFb | Send the DATA DLPDU from station B with T = 1 in the slot reserved by the CTS. |
| | 8 | verify | DLS | INFO_b (s:= add_B; d:= add_A; T:= 1) passed to DLS user | IFb | Verify that the INFO_a message is passed to the DLS user by the station under test. |
| | 9 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 10 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) in ack slot reserved by CTS | IAa | Verify that an ACK is transmitted by station A in the acknowledgement slot reserved by the CTS. |
| | 11 | verify | RF | T = 1 in INFO_ACK_a (s = add_A; d = add_B) | IAa | Verify that the ACK contains T = 1. |
| | 12 | send | RF | INFO_RTS_a (s:= add_B; d:= add_A; T:= 0; pr:= 3; lg:= 5) | IRa | Send an INFO_RTS using the long transmission procedures from a simulated station B, with T = 0. The burst contains a unicast reservation for a response. |
| | 13 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 14 | verify | RF | INFO_CTS_a (s = add_A; d = add_B) in slot reserved by RTS | ICa | Verify that a CTS is transmitted by station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 15 | send | RF | INFO_b (s:= add_B; d:= add_A; T:= 1) in slot reserved by CTS | IFb | Send the DATA DLPDU from station B with T = 1 in the slot reserved by the CTS. This is incorrectly a duplicate INFO burst, since T has the same value as for the last INFO. |
| | 16 | verify | DLS | INFO_b (s:= add_B; d:= add_A; T:= 1) not passed to DLS user | IFb | Verify that the second INFO_a message with T = 1 is not passed to the DLS user by the station under test, and is discarded as a duplicate. |
| | 17 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 18 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) | IAa | Verify that an ACK is transmitted by station A. |
| | 19 | verify | RF | T = 1 in INFO_ACK_a (s = add_A; d = add_B) | IAa | Verify that the ACK contains T = 1. |
| postamble | 20 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Long_Order | | | | |
|-----------------|------|---|-----|---|-------|---|
| Purpose: | | To demonstrate that DATA DLPDUs which are part of the same fragmented packet are concatenated and delivered to the DLS user in the same order in which they appear in the packet. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | M_INITIALIZE_NSCOP | | Initialize the air-ground link. |
| | | Do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | send | RF | INFO_RTS_a (s:= add_B; d:= add_A; pr:= 3; lg:= 5) | IRa | Send an initial RTS to the station under test. The RTS contains a unicast reservation for a response. |
| | 4 | verify | RF | INFO_CTS_a (s = add_A; d = add_B) | ICa | Verify that the station under test sends a CTS in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 5 | rep 5 | | info:= {info1, info2, info3, info4, info5} | | Repeat cycle of RTS / DATA followed by ACK / CTS in order to send a total of 6 fragments of the same packet to the station under test. |
| | 6 | send | RF | INFO_RTS_INFO_b (s:= add_B; d:= add_A; in:= info; INFO (M:= 1; pr:= 3)) in the slot reserved by the CTS | IRIFb | Send an RTS / DATA to the station under test in the slot reserved by the CTS, with the INFO packets containing M = 1. |
| | 7 | verify | RF | INFO_ACK_INFO_CTS_a (s = add_B; d = add_A) in the ack slot reserved by the previous CTS | IAICa | Verify that the station under test sends a CTS / ACK, in the acknowledgement slot reserved by the previous CTS, in response to the RTS / DATA. |
| | 8 | endrep | | next info | | End loop. |
| | 9 | send | RF | INFO_b (s:= add_A; d:= add_B; in:= info6; M:= 0; pr:= 3) in the slot reserved by the CTS | IFb | Send the final INFO burst to the station under test, in the slot reserved by the CTS, with M = 0. |
| | 10 | verify | DLS | Fragments delivered to DLS user concatenated and in the correct order: info1(first), info2, info3, info4, info5, info6 (last) | | Verify that the fragments of the data packet (with the same priority) are delivered to the DLS user of the station under test in the correct order. |
| | 11 | verify | RF | INFO_ACK_a (s = add_B; d = add_A) in the ack slot reserved by the CTS | IAa | Verify that the station under test sends an ACK, in the ack slot reserved by the CTS, in response to the final INFO. |
| postamble | 12 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Long_T_Receive_A | | | | | | |
|--|------|--------|-----|--|-----|--|
| Purpose: To demonstrate that a station correctly sets the T bit when receiving non-combined data packets from another station. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the air-ground link. |
| test body | 4 | send | RF | INFO_RTS_a (s:= add_B; d:= add_A; T:= 1; pr:= 3; lg:= 7) | IRa | Send an INFO_RTS using the long transmission procedures from a simulated station B, with T = 1, indicating this is the first transmission in the sequence from station B to station A following the initialization. The RTS contains a unicast reservation for a response. |
| | 5 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 6 | verify | RF | INFO_CTS_a (s = add_A; d = add_B) transmitted in the slot reserved by the RTS | ICa | Verify that a CTS is transmitted by station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 7 | send | RF | INFO_b (s:= add_B; d:= add_A; T:= 1) in the slot reserved by the CTS | IFb | Send the DATA DLPDU from station B with T = 1 in the slot reserved by the CTS. |
| | 8 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 9 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) transmitted in the ack slot reserved by the CTS | IAa | Verify that an ACK is transmitted by station A in the ack slot reserved by the CTS. |
| | 10 | verify | RF | T = 1 in INFO_ACK_a (s = add_A; d = add_B) | IAa | Verify that the ACK contains T = 1. |
| | 11 | send | RF | INFO_RTS_a (s:= add_B; d:= add_A; T:= 0; pr:= 3; lg:= 4) | IRa | Send the next RTS from station B with T = 0. The RTS contains a unicast reservation for a response. |
| | 12 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 13 | verify | RF | INFO_CTS_a (s = add_A; d = add_B) transmitted in the slot reserved by the RTS | ICa | Verify that a CTS is transmitted by station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 14 | send | RF | INFO_b (s:= add_B; d:= add_A; T:= 0) in the slot reserved by the CTS | IFb | Send the DATA DLPDU from station B with T = 0 in the slot reserved by the CTS. |
| | 15 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 16 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) transmitted in the ack slot reserved by the CTS | IAa | Verify that an ACK is transmitted by station A in the ack slot reserved by the CTS. |
| | 17 | verify | RF | T = 0 in INFO_ACK_a (s = add_A; d = add_B) | IAa | Verify that the ACK contains T = 0. |
| postamble | 18 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Long_T_Receive_B | | | | | | |
|--|------|--------|-----|---|-------|--|
| Purpose: To demonstrate that a station correctly sets the T bit when receiving combined data packets from another station. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the air-ground link. |
| test body | 4 | send | RF | INFO_RTS_a (s:= add_B; d:= add_A; T:= 1; pr:= 3; lg:= 12) | IRa | Send an INFO_RTS using the long transmission procedures from a simulated station B, with T = 1, indicating this is the first transmission in the sequence from station B to station A following the initialization. The RTS contains a unicast reservation for a response. |
| | 5 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 6 | verify | RF | INFO_CTS_a (s = add_A; d = add_B) in the slot reserved by the RTS | ICa | Verify that a CTS is transmitted by station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 7 | send | RF | INFO_RTS_INFO_b (s:= add_B; d:= add_A; INFO (T:= 1); RTS (T:= 0)) in the slot reserved by the CTS | IRIFb | Send an RTS / DATA DLPDU from station B, in the slot reserved by the CTS, with T = 1 in the INFO DLPDU and T = 0 in the RTS. |
| | 8 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 9 | verify | RF | INFO_ACK_INFO_CTS_a (s = add_A; d = add_B) in the ack slot reserved by the CTS | IAICa | Verify that a CTS / ACK is transmitted by station A in the ack slot reserved by the previous CTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 10 | verify | RF | T = 1 in INFO_ACK DLPDU contained in INFO_ACK_INFO_CTS_a (s = add_A; d = add_B) | IAICa | Verify that T = 1 in the INFO_ACK DLPDU in the combined burst. |
| | 11 | send | RF | INFO_RTS_INFO_b (s:= add_B; d:= add_A; INFO (T:= 0); RTS (T:= 1)) in the slot reserved by the CTS | IRIFb | Send an RTS / DATA DLPDU from station B, in the slot reserved by the CTS, with T = 0 in the INFO DLPDU and T = 1 in the RTS. |
| | 12 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 13 | verify | RF | INFO_ACK_INFO_CTS_a (s = add_A; d = add_B) in the ack slot reserved by the CTS | IAICa | Verify that a CTS / ACK is transmitted by station A in the ack slot reserved by the previous CTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 14 | verify | RF | T = 0 in INFO_ACK DLPDU contained in INFO_ACK_INFO_CTS_a (s = add_A; d = add_B) | IAICa | Verify that T = 0 in the INFO_ACK DLPDU and that T = 1 in the INFO DLPDU in the combined burst. |
| | 15 | send | RF | INFO_b (s:= add_B; d:= add_A; T:= 1) in the slot reserved by the CTS | IFb | Send the DATA DLPDU from station B, in the slot reserved by the CTS, with T = 1. |
| | 16 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 17 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) in the ack slot reserved by the CTS | IAa | Verify that an ACK is transmitted by station A in the ack slot reserved by the CTS. |
| | 18 | verify | RF | T = 1 in INFO_ACK_a (s = add_A; d = add_B) | IAa | Verify that the ACK contains T = 1. |
| postamble | 19 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Long_TM_Send_A | | | | |
|-----------------|------|---|-----|--|-----|---|
| Purpose: | | To demonstrate that a station correctly sets the T and M bits when sending a single data packet to another mobile station with the long transmission procedure. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the air-ground link. |
| test body | 4 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d := add_B) with length = 4 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU requiring a length of 4 slots. |
| | 5 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 6 | verify | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Verify that an INFO_RTS is sent by the station under test. The RTS contains a unicast reservation for a response. |
| | 7 | verify | RF | T = 1 in INFO_RTS_a (s = add_A; d = add_B) | IRa | Verify that T = 1 in the RTS, as this is the first transmission in the sequence from station A to station B following the initialization. |
| | 8 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 9 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 10 | verify | RF | INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IFb | Verify that a DATA DLPDU is sent by station A in the slot reserved by the CTS. |
| | 11 | verify | RF | T = 1 and M = 0 in INFO_b (s = add_A; d = add_B) | IFb | Verify that T = 1 and M = 0 in the INFO DLPDU from station A. |
| | 12 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A; T:= 1) in the ack slot reserved by the CTS | IAa | Send the ACK from station B, in the ack slot reserved by the CTS, in response to the completed message with T = 1. |
| postamble | 13 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Long_TM_Send_B | | | | |
|-----------------|------|---|-----|--|-------|---|
| Purpose: | | To demonstrate that a station correctly sets the T and M bits when sending multiple data packets in sequence to another station with the long transmission procedure. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the air-ground link. |
| test body | 4 | rep 3 | | p:= {2, 13, 12} | | Start loop. |
| | 5 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:=add_A; d := add_B; pr:= p) with length = 12 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU requiring a length of 12 slots. |
| | 6 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 7 | verify | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Verify that an INFO_RTS is sent by the station under test, which is the beginning of a long transmission procedure. The RTS contains a unicast reservation for a response. |
| | 8 | verify | RF | T = 1 in INFO_RTS_a (s = add_A; d = add_B) | IRa | Verify that T = 1 in the RTS, as this is the first transmission from station A to station B following the initialization. |
| | 9 | record | | priority(1):= pr in INFO_RTS_a | IRa | Record the priority of the INFO_RTS. |
| | 10 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 11 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 12 | verify | RF | INFO_RTS_INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IRIFb | Verify that an RTS / DATA is sent by station A in the slot reserved by the CTS. |
| | 13 | verify | RF | T = 0 in INFO_RTS DLPDU and T = 1 and M = 1 in INFO DLPDU contained in INFO_RTS_INFO_b (s = add_A; d = add_B) | IRIFb | Verify that T = 0 in the INFO_RTS DLPDU and that T = 1 and M = 1 in the INFO DLPDU in the combined burst. |
| | 14 | record | | priority(2):= pr in INFO_RTS priority(3):= pr in INFO | | Record the priority of the INFO_RTS and of the INFO. |
| | 15 | send | RF | INFO_ACK_INFO_CTS_a (s:= add_B; d:= add_A; INFO_ACK (T:= 1)) in the ack slot reserved by the CTS | IAICa | Send a CTS / ACK from station B in response, in the ack slot reserved by the previous CTS, with T = 1 in the ACK. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 16 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 17 | verify | RF | INFO_RTS_INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IRIFb | Verify that an RTS / DATA is sent by station A in the slot reserved by the CTS. |
| | 18 | verify | RF | T = 1 in INFO_RTS DLPDU and T = 0 and M = 1 in INFO DLPDU contained in INFO_RTS_INFO_b (s = add_A; d = add_B) | IRIFb | Verify that T = 1 in the INFO_RTS DLPDU and that T = 0 and M = 1 in the INFO DLPDU in the combined burst. |
| | 19 | record | | priority(4):= pr in INFO_RTS priority(5):= pr in INFO | | Record the priority of the INFO_RTS and of the INFO. |

| Test case name: DLS_Long_TM_Send_B | | | | | |
|--|------|--------|-----|--|---|
| Purpose: To demonstrate that a station correctly sets the T and M bits when sending multiple data packets in sequence to another station with the long transmission procedure. | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref Comment |
| | 20 | send | RF | INFO_ACK_INFO_CTS_a (s:= add_B; d:= add_A; INFO_ACK (T:= 0)) in the ack slot reserved by the CTS | IAICa Send a CTS / ACK from station B in response, in the ack slot reserved by the previous CTS, with T = 0 in the ACK. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 21 | await | | BURST transmitted by station A | Wait for a burst transmitted by station A. |
| | 22 | verify | RF | INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IFb Verify that a DATA DLPDU is sent by station A in the slot reserved by the CTS. |
| | 23 | verify | RF | T = 1 and M = 0 in INFO_b (s = add_A; d = add_B) | IFb Verify that T = 1 and M = 0 in the INFO DLPDU from station A. |
| | 24 | record | | priority(6):= pr in INFO_b | IFb Record the priority of the INFO. |
| | 25 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A; T:= 1) in the ack slot reserved by the CTS | IAa Send the ACK from station B, in the ack slot reserved by the CTS, in response to the completed message. |
| | 26 | verify | | priority(n) = p for n = 1 to 6 | Verify that the priorities were correctly set in the DLPDUs. |
| | 27 | endrep | | next p | End loop. |
| postamble | 28 | | | | Bring test equipment into idle state. |
| Comments: | | | | | |

| Test case name: | | DLS_Length | | | | |
|-----------------|------|--|-----|--|-------|--|
| Purpose: | | To demonstrate that the length subfield (lg) of an RTS correctly indicates the length in slots of the DLS burst containing a DATA DLPDU. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| test body | 4 | rep 4 | | lg1:= {6, 7, 8} | | Repeat with different values of length. |
| | 5 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B; pr:= 3; lg:= lg1) | | Request that station A send a message to simulated station B that is lg1 slots in length. |
| | 6 | await | RF | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 7 | verify | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Verify the transmission of the RTS from station A. The RTS contains a unicast reservation for a response. |
| | 8 | record | RF | length:= lg from INFO_RTS_a | IRa | Record the length specified in the RTS. |
| | 9 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | ICa | Send a CTS from station B to station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 10 | await | Rf | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 11 | verify | RF | INFO_RTS_INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IRIFb | Verify the transmission of the RTS / INFO from station A in the slot reserved by the CTS. The RTS contains a unicast reservation for a response. |
| | 12 | record | RF | timestart:= time of first slot INFO_RTS_INFO_b is transmitted in | IRIFb | Record the time of the start of the message. |
| | 13 | record | RF | timeend:= time of end of transmission of INFO_RTS_INFO_b | IRIFb | Record the time of the end of the message. |
| | 14 | verify | | timeend - timestart = (length × 60) / 4 500 s | | Verify that the length specified in the RTS is equal to the length of the DATA DLPDU. |
| | 15 | record | RF | length2:= lg in INFO_RTS in combined message | | Record the length specified in the combined RTS. |
| | 16 | send | RF | INFO_ACK_INFO_CTS_a (s:= add_B; d:= add_A) in the ack slot reserved by the CTS | IAICa | Send a CTS / ACK from station B to station A in the ack slot reserved by the previous CTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 17 | await | RF | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 18 | verify | RF | INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IFb | Verify the transmission of the INFO from station A in the slot reserved by the CTS. |
| | 19 | record | RF | timestart:= time of first slot INFO_b is transmitted in | IFb | Record the time of the start of the message. |
| | 20 | record | RF | timeend:= time of end of transmission of INFO_b | IFb | Record the time of the end of the message. |
| | 21 | verify | | timeend - timestart = (length2 × 60) / 4 500 s | | Verify that the length specified in the RTS is equal to the length of the DATA DLPDU. |

| Test case name: DLS_Length | | | | | | |
|---|------|--------|-----|---|-----|---|
| Purpose: To demonstrate that the length subfield (lg) of an RTS correctly indicates the length in slots of the DLS burst containing a DATA DLPDU. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 22 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the ack slot reserved by the CTS | IAa | Send an ACK in response from station B in the ack slot reserved by the CTS. |
| | 23 | next | | lg1 | | Repeat with the next value of lg1. |
| postamble | 24 | | | | | Bring test equipment into idle state. |
| Comments: Is lg still always valid for the combined burst? | | | | | | |

| Test case name: DLS_Priority_Long | | | | | | |
|---|------|--------|-----|---|-------|---|
| Purpose: To demonstrate that a sending station will maintain a prioritized queue of data packets for transmission, and will transmit the highest priority packet first, when data requiring the long transmission procedure is queued to be sent. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| | 3 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 4 | rep 8 | | p:= {2, 3, 5, 7, 12, 6, 8, 13} | | Start loop. |
| | 5 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B; pr:= p) with length 4 slots | | Send requests from a DLS user to the station under test to send a series of 8 DATA DLPDUs with different priorities, all requiring use of the long transmission procedure. |
| | 6 | endrep | | next p | | End loop. |
| | 7 | await | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Await the transmission of the RTS from station A. The RTS contains a unicast reservation for a response. |
| | 8 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 9 | rep 7 | | n:= 1 | | Start loop. |
| | 10 | await | RF | INFO_RTS_INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IRIFb | Wait for the RTS / DATA sent by station A in the slot reserved by the CTS. |
| | 11 | record | RF | priority(n):= pr from INFO_b mbit(n): = value of M in INFO_b | IFb | Record the priority of the DATA and the value of the M bit. |
| | 12 | send | RF | INFO_ACK_INFO_CTS_a (s:= add_B; d:= add_A) in the ack slot reserved by the previous CTS | IAICa | Send a CTS / ACK from station B in response in the ack slot reserved by the previous CTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 13 | endrep | | next n | | End loop. |
| | 14 | await | RF | INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IFb | Wait for the DATA sent by station A in the slot reserved by the CTS. |
| | 15 | record | RF | priority(8):= pr from INFO_b mbit(8): = value of M in INFO_b | IFb | Record the priority of the DATA and the value of the M bit. |
| | 16 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the ack slot reserved by the CTS | IAa | Send an ACK from station B, in the ack slot reserved by the CTS, in response to the completed message. |

| Test case name: DLS_Priority_Long | | | | | | |
|---|------|--------|-----|--|-----|---|
| Purpose: To demonstrate that a sending station will maintain a prioritized queue of data packets for transmission, and will transmit the highest priority packet first, when data requiring the long transmission procedure is queued to be sent. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 17 | verify | | priority(1) = 13 priority(2) = 12 priority(3) = 8 priority(4) = 7 priority(5) = 6 priority(6) = 5 priority(7) = 3 priority(8) = 2 | | Verify that the bursts were transmitted in the order of priority, highest priority being transmitted first. |
| | 18 | verify | | mbit(n) = 0 for n = 1 to 8 | | Verify that the M bit was set to 0 in all DATA DLPDUs. |
| postamble | 19 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Priority_Long_Short | | | | | | |
|--|------|--------|-----|--|-----|---|
| Purpose: To demonstrate that a sending station will maintain a prioritized queue of data packets for transmission, and will transmit the highest priority packet first, when data requiring a mixture of the long and short transmission procedures are queued to be sent. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| | 3 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 4 | rep 8 | | p:= {2, 3, 8, 13} | | Start loop. |
| | 5 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B; pr:= p) with length 4 slots | | Send requests from a DLS user to the station under test to send a series of 4 DATA DLPDUs with different priorities, all requiring use of the long transmission procedure. |
| | 6 | endrep | | next p | | End loop. |
| | 7 | rep 8 | | p:= {1, 14, 9, 14} | | Start loop. |
| | 8 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B; pr:= p) with length 3 slots | | Send requests from a DLS user to the station under test to send a series of 4 DATA DLPDUs with different priorities, all requiring use of the short transmission procedure. |
| | 9 | endrep | | next p | | End loop. |
| | 10 | rep 2 | | n:= 1 | | Start loop. |
| | 11 | await | RF | INFO_a (s = add_A; d = add_B) | IFa | Wait for the first DATA sent by station A. The RTS contains a unicast reservation for a response. |
| | 12 | record | RF | priority(n):= pr from INFO_a mbit(n): = value of M in INFO_a | IFa | Record the priority of the DATA and the value of the M bit. |
| | 13 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the slot reserved by the INFO | IAa | Send an ACK from station B, in the slot reserved by the INFO. |
| | 14 | endrep | | next n | | End loop. |

| Test case name: | | DLS_Priority_Long_Short | | | | |
|-----------------|------|---|-----|--|-------|---|
| Purpose: | | To demonstrate that a sending station will maintain a prioritized queue of data packets for transmission, and will transmit the highest priority packet first, when data requiring a mixture of the long and short transmission procedures are queued to be sent. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 15 | verify | | priority(1) = 14 priority(2) = 14 | | Verify that the correct burst has been sent according to priority order. |
| | 16 | await | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Await the transmission of the RTS from station A. The RTS contains a unicast reservation for a response. |
| | 17 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. |
| | 18 | await | RF | INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IRIFb | Wait for the RTS / DATA sent by station A in the slot reserved by the CTS. |
| | 19 | record | RF | priority(3):= pr from INFO_b mbit(3): = value of M in INFO_b | IFb | Record the priority of the DATA and the value of the M bit. |
| | 20 | verify | | priority(3) = 13 | | Verify that the correct burst has been sent according to priority order. |
| | 21 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the ack slot reserved by the CTS | IAICa | Send an ACK from station B in the ack slot reserved by the CTS. |
| | 22 | await | RF | INFO_a (s = add_A; d = add_B) | IFa | Wait for the DATA sent by station A in the reserved slot. The INFO contains a unicast reservation for a response. |
| | 23 | record | RF | priority(4):= pr from INFO_a mbit(4): = value of M in INFO_a | IFa | Record the priority of the DATA and the value of the M bit. |
| | 24 | verify | | priority(4) = 9 | | Verify that the correct burst has been sent according to priority order. |
| | 25 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the slot reserved by the INFO | IAa | Send an ACK from station B in the slot reserved by the INFO. |
| | 26 | await | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Await the transmission of the RTS from station A. The RTS contains a unicast reservation for a response. |
| | 27 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 28 | await | RF | INFO_RTS_INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IRIFb | Wait for the RTS / DATA sent by station A in the slot reserved by the CTS. The RTS contains a unicast reservation for a response. |
| | 29 | record | RF | priority(5):= pr from INFO_b mbit(5): = value of M in INFO_b | IFb | Record the priority of the DATA and the value of the M bit. |
| | 30 | verify | | priority(5) = 8 | | Verify that the correct burst has been sent according to priority order. |
| | 31 | send | RF | INFO_ACK_INFO_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | IAICa | Send a CTS / ACK from station B in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 32 | await | RF | INFO_RTS_INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IRIFb | Wait for the RTS / DATA sent by station A in the slot reserved by the CTS. The RTS contains a unicast reservation for a response. |
| | 33 | record | RF | priority(6):= pr from INFO_b mbit(6): = value of M in INFO_b | IFb | Record the priority of the DATA and the value of the M bit. |

| Test case name: | | DLS_Priority_Long_Short | | | | |
|-----------------|------|---|-----|---|-------|---|
| Purpose: | | To demonstrate that a sending station will maintain a prioritized queue of data packets for transmission, and will transmit the highest priority packet first, when data requiring a mixture of the long and short transmission procedures are queued to be sent. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 34 | verify | | priority(6) = 3 | | Verify that the correct burst has been sent according to priority order. |
| | 35 | send | RF | INFO_ACK_INFO_CTS_a (s:= add_B; d:= add_A) in the ack slot reserved by the previous CTS | IAICa | Send a CTS / ACK from station B in the ack slot reserved by the previous CTS. |
| | 36 | await | RF | INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IFb | Wait for the DATA sent by station A in the slot reserved by the CTS. |
| | 37 | record | RF | priority(7):= pr from INFO_b mbit(7): = value of M in INFO_b | IFb | Record the priority of the DATA and the value of the M bit. |
| | 38 | verify | | priority(7) = 2 | | Verify that the correct burst has been sent according to priority order. |
| | 39 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the ack slot reserved by the CTS | IAa | Send an ACK from station B in the ack slot reserved by the CTS. |
| | 40 | await | RF | INFO_a (s = add_A; d = add_B) | IFa | Wait for the DATA sent by station A in the reserved slot. The INFO contains a unicast reservation for a response. |
| | 41 | record | RF | priority(8):= pr from INFO_a mbit(8): = value of M in INFO_a | IFa | Record the priority of the DATA and the value of the M bit. |
| | 42 | verify | | priority(8) = 1 | | Verify that the correct burst has been sent according to priority order. |
| | 43 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the slot reserved by the INFO | IAa | Send an ACK from station B in response. |
| | 44 | verify | | mbit(n) = 0 for n = 1 to 8 | | Verify that the M bit was set to 0 in all DATA DLPDUs. |
| postamble | 45 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Priority_Short | | | | |
|-----------------|------|---|-----|---|-----|---|
| Purpose: | | To demonstrate that a sending station will maintain a prioritized queue of data packets for transmission, and will transmit the highest priority packet first, when data requiring the short transmission procedure is queued to be sent. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| | 3 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 4 | rep 8 | | p:= {1, 14, 5, 7, 12, 6, 9, 14} | | Start loop. |
| | 5 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B; pr:= p) with length 3 slots | | Send requests from a DLS user to the station under test to send a series of 8 DATA DLPDUs with different priorities, all requiring use of the short transmission procedure. |
| | 6 | endrep | | next p | | End loop. |
| | 7 | rep 8 | | n:= 1 | | Start loop. |
| | 8 | await | RF | INFO_a (s = add_A; d = add_B) | IFa | Wait for the DATA sent by station A in the reserved slot. The RTS contains a unicast reservation for a response. |
| | 9 | record | RF | priority(n):= pr from INFO_a mbit(n): = value of M in INFO_a | IFa | Record the priority of the DATA and the value of the M bit. |
| | 10 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A) in the slot reserved by the INFO | IAa | Send an ACK from station B in response to the completed message. |
| | 11 | endrep | | n:= n + 1 | | End loop. |
| | 12 | verify | | priority(1) = 14 priority(2) = 14 priority(3) = 12 priority(4) = 9 priority(5) = 7 priority(6) = 6 priority(7) = 5 priority(8) = 1 | | Verify that the bursts were transmitted in the order of priority, highest priority being transmitted first. |
| | 13 | verify | | mbit(n) = 0 for all n values used | | Verify that the M bit was set to 0 in all DATA DLPDUs. |
| postamble | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Priority_CTRL | | | | |
|-----------------|------|---|-----|--|-----|---|
| Purpose: | | To demonstrate that a sending station will classify CTRL and CTRL_RTS as network management messages and assign the highest priority. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | DLS | REQUEST TO TRANSMIT CTRL DLPDU (s:= add_A; d:= add_B; pr:= 2) | | Send a request from a DLS user to the station under test to send a CTRL DLPDU with priority = 2 to a simulated station B. |
| | 3 | await | RF | CTRL_RTS_a (s = add_A; d = add_B; IB:= 1; T:= 0) | CRA | Await the transmission of the RTS from station A. The RTS contains a unicast reservation for a response. |
| | 4 | record | RF | priority:= pr from CTRL_RTS_a | CRA | Record the priority specified in the CTRL_RTS. |
| | 5 | verify | RF | priority = 14 | | Verify that the message has been assigned the highest priority of 14 for network management messages. |
| | 6 | send | RF | CTRL_CTS_a (s:= add_B; d:= add_A; IB:= 1; T:= 0) in the slot reserved by the RTS | CCa | Send a CTS in response from station B to station A in the slot reserved by the RTS. |
| | 7 | await | RF | CTRL_b (s = add_A; d = add_B) in the slot reserved by the CTS | CTb | Wait for the DATA DLPDU sent by station A in the slot reserved by the CTS. |
| | 8 | record | RF | priority:= pr from CTRL_b | CTb | Record the priority specified in the CTRL. |
| | 9 | verify | RF | priority = 14 | | Verify that the message has been assigned the highest priority of 14 for network management messages. |
| | 10 | send | RF | INFO_ACK (s:= add_B; d:= add_A) in the ack slot reserved by the CTS | IAa | Send an ACK in response to the completed message, in the ack slot reserved by the CTS. |
| postamble | 11 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Priority_INFO_RTS | | | | |
|-----------------|------|---|-----|--|-----|---|
| Purpose: | | To demonstrate that an INFO_RTS will be assigned the same priority as the DATA DLPDU with which it is associated. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | M_INITIALIZE_NSCOP | | |
| | 3 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 4 | rep 5 | | p:= {2, 3, 5, 7, 8} | | Start loop. |
| | 5 | send | DLS | REQUEST TO TRANSMIT INFO DLPDU (s:= add_A; d:= add_B; pr:= p) with length 4 slots | | Send a request from a DLS user to the station under test to send an INFO DLPDU with priority = 2, requiring use of the long transmission procedure. |
| | 6 | await | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Await the transmission of the RTS from station A. The RTS contains a unicast reservation for a response. |
| | 7 | record | RF | priority:= pr in INFO_RTS_a | IRa | Record the priority specified in the CTRL_RTS. |
| | 8 | verify | RF | priority = p | | Verify that the message has been assigned the same priority as the INFO burst requested. |
| | 9 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. |
| | 10 | await | RF | INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IFb | Wait for the DATA DLPDU sent by station A in the slot reserved by the CTS. |
| | 11 | record | RF | priority:= pr from INFO_b | IFb | Record the priority specified in the CTRL. |
| | 12 | verify | RF | priority = p | | Verify that the priority is the same as that of the requested burst and also the same as that of the INFO_RTS. |
| | 13 | endrep | | next p | | End loop. |
| | 14 | send | RF | INFO_ACK (s:= add_B; d:= add_A) in the ack slot reserved by the CTS | | Send an ACK in response to the completed message, in the ack slot reserved by the CTS. |
| postamble | 15 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Long_UDATA_Send | | | | |
|-----------------|------|---|-----|--|------|--|
| Purpose: | | To demonstrate that a single UDATA packet is sent correctly using the long transmission procedure, with the UDATA_RTS having the same priority and length fields as the DATA DLPDU with which it is associated. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| | 3 | do | | SET PARAMETERS (ND3:= 7 slots; ND2:= 86 octets) | | Set the parameter ND3 = 7 slots and ND2 = 86 octets (3 slots). |
| test body | 4 | rep 4 | | p:= {2, 3, 5, 8}, l1:= {4, 5, 6, 7} | | Start loop. |
| | 5 | send | DLS | REQUEST TO TRANSMIT UDATA DLPDU (s:= add_A; d:= add_B; pr:= p) with length l1 slots | | Send a request from a DLS user to the station under test to send a UDATA DLPDU requiring use of the long transmission procedure. |
| | 6 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 7 | verify | RF | UDATA_RTS_a (s = add_A; d = add_B) | URa | Verify that a UDATA_RTS is transmitted by station A. The UDATA_RTS contains a unicast reservation for a response. |
| | 8 | record | RF | priority:= pr length:= lg in UDATA_RTS_a | URa | Record the priority and the length specified in the UDATA_RTS. |
| | 9 | verify | RF | priority = p length = l1 | | Verify that the message has been assigned the same priority and length as the UDATA burst requested. |
| | 10 | send | RF | UDATA_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the UDATA_RTS | UDCa | Send a UDATA_CTS in response from station B to station A, in the slot reserved by the UDATA_RTS. The UDATA_CTS contains a unicast reservation that also includes a reservation for an acknowledgement. |
| | 11 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 12 | verify | RF | UINFO_a (s = add_A) in the slot reserved by the UDATA_CTS | UIa | Verify that a UINFO DLPDU is broadcast by station A in the reserved slot. |
| postamble | 13 | endrep | | next p, l1 | | End loop. |
| | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Initialize_Send_Long | | | | |
|-----------------|------|---|-----|--|-----|--|
| Purpose: | | To demonstrate that a station displays correct behaviour and correctly sets the T and IB bits when sending data packets to another station during link initialization with the long transmission procedure. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | send | LME | REQUEST TO TRANSMIT CTRL DLPDU (s:= add_A; d := add_B) with length = 4 slots | | Send a request from the LME to the station under test to send a DATA DLPDU requiring a length of 4 slots. |
| | 4 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_RTS_a (s = add_A; d = add_B) | CRA | Verify that a CTRL_RTS DLPDU is sent by the station under test, indicating the start of a long transmission procedure. The RTS contains a unicast reservation for a response. |
| | 6 | verify | RF | IB = 1 and T = 0 and lg = 4 in CTRL_RTS_a (s = add_A; d = add_B) | CRA | Verify that IB = 1 and T = 0 and lg = 4 in the CTRL_RTS, as this is the first transmission from station A to station B. |
| | 7 | send | RF | CTRL_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | CCA | Send a CTS in response from station B to station A in the slot reserved by the RTS. |
| | 8 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 9 | verify | RF | CTRL_b (s = add_A; d = add_B) in the slot reserved by the CTS | CTb | Verify that the station under test sends a CTRL DLPDU to station B in the slot reserved by the CTS. |
| | 10 | verify | RF | T = 0 in CTRL_b DLPDU | CTb | Verify that T = 0 in the CTRL DLPDU. |
| | 11 | send | RF | CTRL_ACK_a (s:= add_B; d:= add_A; T:= 0) in the ack slot reserved by the CTS | CAa | Send the ACK from station B in response to the completed message in the ack slot reserved by the CTS. |
| | 12 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 0) | IFa | Send an INFO DLPDU to the station under test. The INFO contains a unicast reservation for a response. |
| | 13 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 14 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an INFO_ACK, in the slot reserved by the INFO, and not a DM / DISC, thus indicating that the station under test considers the link established. |
| postamble | 15 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_INFO_RTS_SZOM_Send | | | | | | |
|--|------|--------|-----|--|-------|--|
| Purpose: To demonstrate that a station displays correct behaviour and correctly sets the T bit when sending data packets to another station during ZOCOP link initialization with the long transmission procedure. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | send | LME | REQUEST TO TRANSMIT INFO DLPDU (s:= add_A; d := add_B) with length = 4 slots | | Send a request from the LME to the station under test to send an INFO DLPDU requiring a length of 4 slots. |
| | 4 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | INFO_RTS_SZOM_a (s = add_A; d = add_B) | IRSZa | Verify that an INFO_RTS_SZOM is sent by the station under test, indicating the start of a long transmission procedure. The RTS_SZOM contains a unicast reservation for a response. |
| | 6 | verify | RF | T = 0, neg = 0, and lg = 4 in INFO_RTS_SZOM_a (s = add_A; d = add_B) | IRSZa | Verify that T = 0, neg = 0, and lg = 4 in the INFO_RTS_SZOM. T = 0 as this is the first transmission from station A to station B. |
| | 7 | send | RF | INFO_CTS_SZOM_a (s:= add_B; d:= add_A; neg:= 0; seq:= 0) in the slot reserved by the RTS | ICSZa | Send an INFO_CTS_SZOM in response from station B to station A in the slot reserved by the RTS. The neg and seq subfields are set to zero. |
| | 8 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 9 | verify | RF | INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IFb | Verify that the station under test sends the INFO to station B in the slot reserved by the CTS. |
| | 10 | verify | RF | T = 0 and M = 0 in INFO_b (s = add_A; d = add_B) | IFb | Verify that T = 0 and M = 0 in the INFO DLPDU. |
| | 11 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A; T:= 0) in the ack slot reserved by the CTS | IAa | Send the ACK from station B in response to the completed message in the ack slot reserved by the CTS. |
| | 12 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 0) | IFa | Send an INFO to the station under test. The INFO contains a unicast reservation for a response. |
| | 13 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 14 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an INFO_ACK, in the slot reserved by the INFO, and not a DM / DISC, thus indicating that the station under test considers the link established. |
| postamble | 15 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_INFO_SZOM_Send | | | | |
|------------------|------|--|-----|---|-------|--|
| Purpose: | | To demonstrate that a station displays correct behaviour and correctly sets the T bit when sending data packets to another station during ZOCOP link initialization with the short transmission procedure. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | send | LME | REQUEST TO TRANSMIT INFO DLPDU (s:= add_A; d := add_B) with length = 3 slots | | Send a request from the LME to the station under test to send an INFO DLPDU requiring a length of 3 slots. |
| | 4 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Verify that an INFO_SZOM is sent by the station under test, indicating a short transmission procedure. The INFO_SZOM contains a unicast reservation for a response. |
| | 6 | verify | RF | T = 0, M = 0 in INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Verify that T = 0 and M = 0 in the INFO_SZOM. T = 0 as this is the first transmission from station A to station B. |
| | 11 | send | RF | INFO_ACK_SZOM_a (s:= add_B; d:= add_A; T:= 0; neg:= 0; seq:= 0) in the ack slot reserved by the INFO_SZOM | IASZa | Send an INFO_ACK_SZOM from station B in response to the INFO_SZOM. |
| | 12 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 0) | IFa | Send an INFO DLPDU to the station under test. The INFO contains a unicast reservation for a response. |
| | 13 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 14 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an INFO_ACK, in the slot reserved by the INFO, and not a DM / DISC, thus indicating that the station under test considers the link established. |
| postamble | 15 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_INFO_RTS_SZOM_Receive | | | | |
|-----------------|------|---|-----|---|-------|---|
| Purpose: | | To demonstrate that a station displays correct behaviour and correctly sets the T bit when receiving data packets from another station during ZOCOP link initialization with the long transmission procedure. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | send | RF | INFO_RTS_SZOM_a (s:= add_B; d:= add_A; T:= 0; neg:= 0; lg:= 4) | IRSZa | Send an INFO_RTS_SZOM using the long transmission procedures from a simulated station B, with T = 0, neg = 0 and lg = 4, indicating this is the first transmission from station B to station A. The RTS_SZOM contains a unicast reservation for a response. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | INFO_CTS_SZOM_a (s = add_A; d = add_B) in the slot reserved by the RTS | ICSZa | Verify that a CTRL_CTS is sent by the station under test in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 6 | verify | RF | neg = 0 and seq = 0 in the INFO_CTS_SZOM_a (s:= add_B; d:= add_A) | ICSZa | Verify that neg = 0 and seq = 0 in the INFO_CTS_SZOM. |
| | 7 | send | RF | INFO_b (s:= add_B; d:= add_A; T:= 0; M:= 0) in the slot reserved by the CTS | IFb | Send an INFO from station B with T = 0 and M = 0 in the slot reserved by the CTS. |
| | 8 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 9 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) in the ack slot reserved by the CTS | IAa | Verify that an INFO_ACK is sent by the station under test in the ack slot reserved by the CTS. |
| | 10 | verify | RF | T = 0 in INFO_ACK_a (s = add_A; d = add_B) | IAa | Verify that the INFO_ACK contains T = 0. |
| | 11 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 0) | IFa | Send an INFO DLPDU to the station under test. The INFO contains a unicast reservation for a response. |
| | 12 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 13 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an INFO_ACK, in the slot reserved by the INFO, and not a DM / DISC, thus indicating that the station under test considers the link established. |
| postamble | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_INFO_SZOM_Receive | | | | |
|-----------------|------|--|-----|---|-------|---|
| Purpose: | | To demonstrate that a station displays correct behaviour and correctly sets the T bit when receiving data packets from another station during ZOCOP link initialization with the short transmission procedure. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | send | RF | INFO_SZOM_a (s:= add_B; d:= add_A; T:= 0; neg:= 0; lg:= 4) | IFSZa | Send an INFO_SZOM using the short transmission procedures from a simulated station B, with T = 0, neg = 0 and lg = 4, indicating this is the first transmission from station B to station A. The INFO_SZOM contains a unicast reservation for a response. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | INFO_ACK_SZOM_a (s = add_A; d = add_B) in the slot reserved by the INFO_SZOM | IASZa | Verify that an INFO_ACK_SZOM is sent by the station under test in the slot reserved by the INFO_SZOM. |
| | 10 | verify | RF | T = 0, neg = 0 and seq = 0 in INFO_ACK_SZOM_a (s = add_A; d = add_B) | IASZa | Verify that the INFO_ACK_SZOM contains T = 0, neg = 0 and seq = 0. |
| | 11 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 0) | IFa | Send an INFO DLPDU to the station under test. The INFO contains a unicast reservation for a response. |
| | 12 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 13 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an INFO_ACK, in the slot reserved by the INFO, and not a DM / DISC, thus indicating that the station under test considers the link established. |
| postamble | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_TD1_Link | | | | |
|-----------------|------|---|-----|---|-------|--|
| Purpose: | | To demonstrate that a station that has established a ZOCOP link with a peer, and which has not sent a packet to that peer for TD1 seconds, and that wishes to send a DATA packet to the peer, will re-establish the link. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | rep 3 | | t1:= {40, 60, 80} | | Repeat for different values of TD1. |
| | 4 | do | | SET PARAMETERS (TD1:= t1) | | Set the value of the parameter TD1. |
| | 5 | do | | M_INITIALIZE_ZOCOP | | Initialize the air-air link. |
| | 6 | wait | RF | t1 seconds | | Wait until TD1 has expired. |
| | 7 | send | LME | REQUEST TO TRANSMIT INFO DLPDU (s:= add_A; d := add_B) with length = 3 slots | | Send a request from the LME to the station under test to send an INFO DLPDU requiring a length of 3 slots. |
| | 8 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 9 | verify | RF | INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Verify that an INFO_SZOM is sent by the station under test, indicating a short transmission procedure. The INFO_SZOM contains a unicast reservation for a response. |
| | 10 | verify | RF | T = 0, M = 0 in INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Verify that T = 0 and M = 0 in the INFO_SZOM. T = 0 as this is the first transmission from station A to station B. |
| | 11 | send | RF | INFO_ACK_SZOM_a (s:= add_B; d:= add_A; T:= 0; neg:= 0; seq:= 0) in the ack slot reserved by the INFO_SZOM | IASZa | Send an INFO_ACK_SZOM from station B in response to the INFO_SZOM. |
| | 12 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 0) | IFa | Send an INFO DLPDU to the station under test. The INFO contains a unicast reservation for a response. |
| | 13 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 14 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an INFO_ACK, in the slot reserved by the INFO, and not a DM / DISC, thus indicating that the station under test considers the link established. |
| | 15 | endrep | | next t1 | | End loop. |
| postamble | 16 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_TD1_TD2_Link | | | | |
|-----------------|------|---|-----|---|-------|---|
| Purpose: | | To demonstrate that a station will regard the link as terminated if it has not transmitted to the same peer for TD1 seconds and if it has not received a transmission from the same peer for TD2 seconds. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | rep 3 | | t1:= {40, 80, 60}; t2:= {80, 30, 90} | | Repeat for different values of TD1 and TD2. |
| | 4 | do | | SET PARAMETERS (TD1:= t1; TD1:= t2) | | Set the value of the parameter TD1. |
| | 5 | do | | M_INITIALIZE_ZOCOP | | Initialize the air-air link. |
| | 6 | wait | RF | MAX(t1, t2) seconds | | Wait until TD1 and TD2 have expired. |
| | 7 | send | LME | REQUEST TO TRANSMIT INFO DLPDU (s:= add_A; d := add_B) with length = 3 slots | | Send a request from the LME to the station under test to send an INFO DLPDU requiring a length of 3 slots. |
| | 8 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 9 | verify | RF | INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Verify that an INFO_SZOM is sent by the station under test, indicating a short transmission procedure. The INFO_SZOM contains a unicast reservation for a response. |
| | 10 | verify | RF | T = 0, M = 0 in INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Verify that T = 0 and M = 0 in the INFO_SZOM. T = 0 as this is the first transmission from station A to station B. |
| | 11 | send | RF | INFO_ACK_SZOM_a (s:= add_B; d:= add_A; T:= 0; neg:= 0; seq:= 0) in the ack slot reserved by the INFO_SZOM | IASZa | Send an INFO_ACK_SZOM from station B in response to the INFO_SZOM. |
| | 12 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 0) | IFa | Send an INFO DLPDU to the station under test. The INFO contains a unicast reservation for a response. |
| | 13 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 14 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an INFO_ACK in the slot reserved by the INFO, and not a DM / DISC, thus indicating that the station under test considers the link established. |
| | 15 | endrep | | next t1, t2 | | End loop. |
| postamble | 16 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Link_Terminated | | | | | | |
|--|------|--------|-----|---|-------------|---|
| Purpose: To demonstrate that a station will regard the link as terminated if it receives a DM/DISC DLPDU or a DM/FRMR from the peer station. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | rep 2 | | dis:= {disc, frmr} | | Repeat for different values of TD1. |
| | 4 | do | | M_INITIALIZE_ZOCOP | | Initialize the air-air link. |
| | 5 | send | RF | IF dis = disc THEN DM_DISC_a (s:= add_B; d:= add_A) ELSE IF dis = frmr THEN DM_FRMR_a (s:= add_B; d:= add_A) | DDa, DFa | Send a DM/DISC or DM/FRMR from station B to the station under test. |
| | 6 | send | LME | REQUEST TO TRANSMIT INFO DLPDU (s:= add_A; d := add_B) with length = 3 slots | | Send a request from the LME to the station under test to send an INFO DLPDU requiring a length of 3 slots. |
| | 7 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 8 | verify | RF | INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Verify that an INFO_SZOM is sent by the station under test, indicating a short transmission procedure. The INFO_SZOM contains a unicast reservation for a response. |
| | 9 | verify | RF | T = 0, M = 0 in INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Verify that T = 0 and M = 0 in the INFO_SZOM. T = 0 as this is the first transmission from station A to station B. |
| | 10 | send | RF | INFO_ACK_SZOM_a (s:= add_B; d:= add_A; T:= 0; neg:= 0; seq:= 0) in the ack slot reserved by the INFO_SZOM | IASZa | Send an INFO_ACK_SZOM from station B in response to the INFO_SZOM. |
| | 11 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 0) | IFa | Send an INFO DLPDU to the station under test. The INFO contains a unicast reservation for a response. |
| | 12 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 13 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an INFO_ACK in the slot reserved by the INFO, and not a DM / DISC, thus indicating that the station under test considers the link established. |
| | 14 | endrep | | next dis | | End loop. |
| postamble | 15 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_SZOM_Retrans | | | | |
|-----------------|------|--|-----|---|------------------|---|
| Purpose: | | To demonstrate that a station which transmitted an SZOM to a peer station will retransmit the SZOM in response to any DLPDU other than an SZOM, DM/DISC, DM/FRMR, or general failure, until it receives an SZOM. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | rep 3 | | burstT:= {info, ack, rts} | | Repeat for different bursts. |
| | 4 | send | DLS | REQUEST TO TRANSMIT INFO DLPDU (s:= add_A; d:= add_B) with length requiring burst = 3 slots | | Send a request from a DLS user to the station under test to send an INFO DLPDU to a simulated station B requiring a burst of length 11 slots. |
| | 5 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 6 | verify | RF | INFO_SZOM_a (s = add_A; d = add_B) transmitted | IFSZa | Verify that an INFO_SZOM is sent by the station under test, indicating a short transmission procedure. The INFO_SZOM contains a unicast reservation for a response. |
| | 7 | record | RF | timedata1:= time at beginning of slot containing INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Record the time that the INFO_SZOM was transmitted by the station under test. |
| | 8 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 9 | verify | RF | INFO_SZOM_a (s = add_A; d = add_B) transmitted before time:= timedata1 + 2,45 | IFSZa | Verify that the station under test attempts to retransmit the data. The INFO_SZOM contains a unicast reservation for a response. |
| | 10 | record | RF | timedata2:= time at beginning of slot containing INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Record the time that the INFO_SZOM was retransmitted by the station under test. |
| | 11 | send | RF | IF burst = info THEN INFO_a (s:= add_B; d:= add_A) ELSE IF burst = ack THEN INFO_ACK_a (s:= add_B; d:= add_A) ELSE IF burst = rts THEN INFO_RTS_a (s:= add_B; d:= add_A) | IFa, Iaa, IRa | Send a burst to the station under test that is not an SZOM, DM/DISC, DM/FRMR, or general failure. The station under test ignores the burst. |
| | 12 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 13 | verify | RF | INFO_SZOM_a (s = add_A; d = add_B) transmitted before time:= timedata2 + 3,465 | IFSZa | Verify that the station under test attempts to retransmit the data. The INFO_SZOM contains a unicast reservation for a response. |
| | 14 | record | RF | timedata3:= time at beginning of slot containing INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Record the time that the INFO_SZOM was retransmitted by the station under test. |
| | 15 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 16 | verify | RF | INFO_SZOM_a (s = add_A; d = add_B) transmitted before time:= timedata3 + 4,1905 | IFSZa | Verify that the station under test attempts to retransmit the data. The INFO_SZOM contains a unicast reservation for a response. |
| | 17 | record | RF | timedata4:= time at beginning of slot containing INFO_SZOM_a (s = add_A; d = add_B) | IFSZa | Record the time that the INFO_SZOM was retransmitted by the station under test. |
| | 18 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 19 | verify | RF | INFO_SZOM_a (s = add_A; d = add_B) transmitted before time:= timedata4 + 8,12385 | IFSZa | Verify that the station under test attempts to retransmit the data. The INFO_SZOM contains a unicast reservation for a response. |
| | 20 | verify | DLS | notification received that Q5num attempts have been made to transmit INFO_SZOM_a with no ACK | IFSZa | Verify that the station notifies the DLS that Q5num attempts have been made and that no ACK was received following any of the attempts. |

| Test case name: DLS_SZOM_Retrans | | | | | | |
|---|------|--------|-----|---|---------------|---|
| Purpose: To demonstrate that a station which transmitted an SZOM to a peer station will retransmit the SZOM in response to any DLPDU other than an SZOM, DM/DISC, DM/FRMR, or general failure, until it receives an SZOM. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 21 | send | RF | IF burst = info THEN INFO_a (s:= add_B; d:= add_A) ELSE IF burst = ack THEN INFO_ACK_a (s:= add_B; d:= add_A) ELSE IF burst = rts THEN INFO_RTS_a (s:= add_B; d:= add_A) | IFa, IAa, IRa | Send a burst to the station under test that is not an SZOM, DM/DISC, DM/FRMR, or general failure. |
| | 22 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 23 | verify | RF | DM_DISC_a (s:= add_A; d:= add_B) | DDa | Verify that a DM/DISC is transmitted by station A. |
| | 24 | endrep | | next burst | | End loop. |
| postamble | 25 | do | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_ZOCOP_Link_CTRL | | | | | | |
|---|------|--------|-----|---|--------------------|---|
| Purpose: To demonstrate that if a CTRL_RTS (IB = 0), CTRL_CTS, CTRL, or CTRL_ACK DLPDU is received on an established ZOCOP link, then the receiver shall respond with a DM/FRMR in the reserved slot. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | rep 2 | | burst:= {rts, cts, ctrl, cack) | | Repeat for different values of TD1. |
| | 4 | do | | M_INITIALIZE_ZOCOP | | Initialize the air-air link. |
| | 5 | send | RF | IF burst = rts THEN CTRL_RTS_a (s:= add_B; d:= add_A; IB:= 0) ELSE IF burst = cts THEN CTRL_CTS_a (s:= add_B; d:= add_A) ELSE IF burst = cts THEN CTRL_a (s:= add_B; d:= add_A) ELSE IF burst = cts THEN CTRL_ACK_a (s:= add_B; d:= add_A) | CRa, CCa, CTa, CAa | Send a CTRL_RTS (IB = 0), CTRL_CTS, CTRL, or CTRL_ACK DLPDU from station B to the station under test. |
| | 6 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 7 | verify | RF | DM_FRMR_a (s = add_A; d = add_B) in slot reserved by station B (except if burst = cack) | DFa | Verify that a DM/FRMR is sent by the station under test. |
| | 8 | endrep | | next burst | | End loop. |
| postamble | 9 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Initialize_Receive_Long | | | | | | |
|--|------|--------|-----|---|-----|---|
| Purpose: To demonstrate that a station displays correct behaviour and correctly sets the T bit when receiving data packets from another station during link initialization with the long transmission procedure. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | send | RF | CTRL_RTS_a (s:= add_B; d:= add_A; IB:= 1; T:= 0; lg:= 4) | CRa | Send an RTS using the long transmission procedures from a simulated station B, with IB = 1 and T = 0, indicating this is the first transmission from station B to station A. The RTS contains a unicast reservation for a response. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_CTS_a (s = add_A; d = add_B) in the slot reserved by the RTS | CCa | Verify that a CTRL_CTS is sent by the station under test in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 6 | send | RF | CTRL_b (s:= add_B; d:= add_A; T:= 0) in the slot reserved by the CTS | CTb | Send a CTRL DLPDU from station B with T = 0 in the slot reserved by the CTS. |
| | 7 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 8 | verify | RF | CTRL_ACK_a (s = add_A; d = add_B) in the ack slot reserved by the CTS | CAa | Verify that a CTRL_ACK is sent by the station under test in the ack slot reserved by the CTS. |
| | 9 | verify | RF | T = 0 in CTRL_ACK_a (s = add_A; d = add_B) | CAa | Verify that the CTRL_ACK contains T = 0. |
| | 10 | send | RF | INFO_a (s:= add_B; d:= add_A; T:= 0) | IFa | Send an INFO DLPDU to the station under test. The INFO contains a unicast reservation for a response. |
| | 11 | await | | BURST from station under test | | Wait for a burst from the station under test. |
| | 12 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an INFO_ACK, in the slot reserved by the INFO and not a DM / DISC, thus indicating that the station under test considers the link established. |
| postamble | 13 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Short_T_Receive | | | | | | |
|--|------|--------|-----|--|-----|--|
| Purpose: To demonstrate that a station will respond with an ACK with the T bit set correctly when it receives a DATA DLPDU via the short transmission procedure. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| test body | 4 | send | RF | INFO_a (s = add_B; d = add_A; T = 1) | IFa | Send a DATA DLPDU by the short transmission procedure to the station under test with T = 1. The INFO contains a unicast reservation for a response. |
| | 5 | verify | DLS | Data in INFO_a (s:= add_B; d = add_A) passed to DLS user | IFa | Verify that the data in the DATA DLPDU is passed to the DLS user. |
| | 6 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 7 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an ACK in response, in the slot reserved by the INFO. |
| | 8 | verify | RF | T = 1 in INFO_ACK | IAa | Verify that T = 1 in the ACK. |
| | 9 | send | RF | INFO_a (s = add_B; d = add_A; T = 0) | IFa | Send a DATA DLPDU by the short transmission procedure to the station under test with T = 0. The INFO contains a unicast reservation for the response |
| | 10 | verify | DLS | Data in INFO_a (s:= add_B; d = add_A) passed to DLS user | IFa | Verify that the data in the DATA DLPDU is passed to the DLS user. |
| | 11 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 12 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an ACK in response, in the slot reserved by the INFO. |
| postamble | 13 | verify | RF | T = 0 in INFO_ACK | IAa | Verify that T = 0 in the ACK. |
| | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Short_T_Send | | | | |
|-----------------|------|--|-----|--|-----|--|
| Purpose: | | To demonstrate that a station correctly sets the T and M bits when sending a single data packet to another mobile station with the short transmission procedure. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the air-ground link. |
| test body | 4 | rep 3 | | l1:= {1, 2, 3}; p:= {3, 14, 11} | | Start loop. |
| | 5 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d := add_B; pr:= p) with length = l1 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU requiring a length of l1 slots. |
| | 6 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 7 | verify | RF | INFO_a (s = add_A; d = add_B) | IFa | Verify that an INFO DLPDU is sent by the station under test. The INFO contains a unicast reservation for a response. |
| | 8 | verify | RF | T = 1 in INFO_a (s = add_A; d = add_B) | IFa | Verify that T = 1 in the INFO DLPDU, as this is the first transmission in the sequence from station A to station B following the initialization. |
| | 9 | verify | RF | M = 0 in INFO_a (s = add_A; d = add_B) | IFa | Verify that M = 0 as required for a short transmission. |
| | 10 | verify | RF | pr = p in INFO_a (s = add_A; d = add_B) | IFa | Verify that the priority pr = p. |
| | 11 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A; T:= 1) in the slot reserved by the INFO | IAa | Send the ACK from station B, in the slot reserved by the INFO, in response to the completed message with T = 1. |
| | 12 | endrep | | next l1, next p | | End loop. |
| postamble | 13 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Short_DATA_ACK | | | | |
|-----------------|------|--|-----|--|-------|--|
| Purpose: | | To demonstrate that a station which has data to send back to the sending station that would fit into a single slot with the ACK, will include its own data DLPDU with the ACK. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link between station B and the station under test. |
| test body | 3 | send | RF | INFO_a (s = add_B; d = add_A; T = 1) | IFa | Send a DATA DLPDU by the short transmission procedure to the station under test with T = 1. The INFO contains a unicast reservation for a response. |
| | 4 | send | DLS | At the same time as the start of the slot in the previous step REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d := add_B; pr:= 3) with (length + ACK) < 1 slot | | Send a request from a DLS user to the station under test to send a DATA DLPDU that together with the ACK will fit into 1 slot. Send this request at the start of the slot containing the INFO_a burst. |
| | 5 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 6 | verify | RF | INFO_ACK_INFO_a (s = add_A; d = add_B) in the slot reserved by the INFO | IAIFa | Verify that the station under test sends an ACK combined with an INFO in the slot reserved by the previous INFO. The INFO contains a unicast reservation for a response. |
| | 7 | verify | RF | T = 1 in INFO_ACK and T = 0 in INFO in INFO_ACK_INFO_a (s = add_A; d = add_B) | IAIFa | Verify that T = 1 in the ACK and that T = 0 in the INFO DLPDU since this is the first DATA transmission from A to B. |
| | 8 | send | RF | INFO_ACK_a (s = add_B; d = add_A; T = 0) in the slot reserved by the INFO | IFa | Send an ACK from station B to the station under test with T = 0, in the slot reserved by the INFO. |
| postamble | 9 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Long_DATA_ACK | | | | | | |
|--|------|--------|-----|---|-------|---|
| Purpose: To demonstrate that a station which has data that would not fit into a single slot with the ACK, will send back to the sending station an RTS DLPDU with the ACK. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link between station B and the station under test. |
| test body | 3 | send | RF | INFO_a (s = add_B; d = add_A; T = 1) | IFa | Send a DATA DLPDU by the short transmission procedure to the station under test with T = 1. The INFO contains a unicast reservation for a response. |
| | 4 | send | DLS | At the same time as the start of the slot in the previous step REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d := add_B; pr:= 3) with length = 2 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU requiring 2 slots. Send this request at the start of the slot containing the INFO_a burst. |
| | 5 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 6 | verify | RF | INFO_ACK_INFO_RTS_a (s = add_A; d = add_B) in the slot reserved by the INFO | IAIRa | Verify that the station under test sends an ACK combined with an RTS in the slot reserved by the INFO. The RTS contains a unicast reservation for a response. |
| | 7 | verify | RF | T = 1 in INFO_ACK and T = 0 in INFO_RTS in INFO_ACK_INFO_RTS_a (s = add_A; d = add_B) | IAIRa | Verify that T = 1 in the ACK and that T = 0 in the RTS DLPDU since this is the first DATA transmission from A to B. |
| | 8 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 9 | await | RF | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 10 | verify | RF | INFO_b (s = add_A; d = add_B) in the slot reserved by the CTS | IFb | Verify that a DATA DLPDU is sent by station A in the slot reserved by the CTS. |
| | 11 | verify | RF | T = 0 and M = 0 in INFO_b (s = add_A; d = add_B) | IFb | Verify that T = 0 and M = 0 in the INFO DLPDU from station A. |
| | 12 | send | RF | INFO_ACK_a (s:= add_B; d:= add_A; T:= 0) in the ack slot reserved by the CTS | IAa | Send the ACK from station B in response to the completed message with T = 0, in the ack slot reserved by the CTS. |
| postamble | 13 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Short_NoAck | | | | |
|-----------------|------|---|-----|--|-----|---|
| Purpose: | | To demonstrate that a station will retransmit the DLS burst, according to the default DLS retransmission parameters, if an acknowledgement is not received. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| test body | 4 | rep 3 | | I1:= {1, 2, 3} | | Start loop. |
| | 5 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B) with length requiring burst = I1 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU to a simulated station B requiring a burst of length I1 slots. |
| | 6 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 7 | verify | RF | INFO_a (s = add_A; d = add_B) transmitted | IFa | Verify that the station under test sends a DATA DLPDU by the short transmission procedures. The INFO contains a unicast reservation for a response. |
| | 8 | record | RF | timedata1:= time at beginning of slot containing INFO_a (s = add_A; d = add_B) | IFa | Record the time that the DATA was transmitted by the station under test. |
| | 10 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 11 | verify | RF | INFO_a (s = add_A; d = add_B) transmitted before time:= timedata1 + 2,45 | IFa | Verify that the station under test attempts to retransmit the data. The INFO contains a unicast reservation for a response. |
| | 12 | record | RF | timedata2:= time at beginning of slot containing INFO_a (s = add_A; d = add_B) | IFa | Record the time that the DATA was retransmitted by the station under test. |
| | 13 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 14 | verify | RF | INFO_a (s = add_A; d = add_B) transmitted before time:= timedata 2 + 3,465 | IFa | Verify that the station under test attempts to retransmit the data. The INFO contains a unicast reservation for a response. |
| | 15 | record | RF | timedata3:= time at beginning of slot containing INFO_a (s = add_A; d = add_B) | IFa | Record the time that the DATA was retransmitted by the station under test. |
| | 16 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 17 | verify | RF | INFO_a (s = add_A; d = add_B) transmitted before time:= timedata3 + 4,1905 | IFa | Verify that the station under test attempts to retransmit the data. The INFO contains a unicast reservation for a response. |
| | 18 | record | RF | timedata4:= time at beginning of slot containing INFO_a (s = add_A; d = add_B) | IFa | Record the time that the DATA was retransmitted by the station under test. |
| | 19 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 20 | verify | RF | INFO_a (s = add_A; d = add_B) transmitted before time:= timedata4 + 8,12385 | IFa | Verify that the station under test attempts to retransmit the data. The INFO contains a unicast reservation for a response. |
| | 21 | verify | DLS | notification received that Q5num attempts have been made to transmit INFO_a with no ACK | | Verify that the station notifies the DLS that Q5num attempts have been made and that no ACK was received following any of the attempts. |
| | 22 | do | | SET PARAMETERS (Q5wait:= 1) | | Set the parameter Q5wait to the minimum value. |
| | 23 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d:= add_B) with length requiring burst = I1 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU to a simulated station B requiring a burst of length I1 slots. |
| | 24 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |

| Test case name: DLS_Short_NoAck | | | | | | |
|--|------|--------|-----|---|-----|---|
| Purpose: To demonstrate that a station will retransmit the DLS burst, according to the default DLS retransmission parameters, if an acknowledgement is not received. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 25 | verify | RF | INFO_a (s = add_A; d = add_B) transmitted | IFa | Verify that the station under test sends a DATA DLPDU by the short transmission procedures. The INFO contains a unicast reservation for a response. |
| | 26 | record | RF | timedata:= time at beginning of slot containing INFO_a (s = add_A; d = add_B) | IFa | Record the time that the DATA was transmitted by the station under test. |
| | 27 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 28 | verify | RF | INFO_a (s = add_A; d = add_B) transmitted before time:= timedata + 1 | IFa | Verify that the station under test attempts to retransmit the data. The INFO contains a unicast reservation for a response. |
| | 29 | verify | DLS | notification received that the station has waited more than Q5wait seconds to transmit INFO_a with no ACK | | Verify that the station notifies the DLS that it has waited more than Q5 wait seconds, and that no ACK was received. |
| postamble | 30 | do | | SET PARAMETERS (Q5wait:= 60) | | Reset the parameter Q5wait to the default value. |
| Comments: | | | | | | |

| Test case name: DLS_Long_UDATA_Receive | | | | | | |
|---|------|--------|-----|--|------|---|
| Purpose: To demonstrate that a station will correctly issue a CTS for an RTS_UDATA received via the long transmission procedure from another station. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| test body | 3 | send | RF | UDATA_RTS_a (s = add_B; d = add_A; pr:= 3; lg:= 4) | URa | Send a UDATA_RTS DLPDU to the station under test. The UDATA_RTS contains a unicast reservation for a response. |
| | 4 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 5 | verify | RF | UDATA_CTS_a (s:= add_A; d:= add_B) in the slot reserved by the UDATA_RTS | UDCa | Verify that a UDATA_CTS is transmitted by station A in the slot reserved by the UDATA_RTS. The burst contains a unicast request reservation for a response. |
| | 6 | send | RF | UINFO_a (s = add_B) in the slot reserved by the UDATA_CTS | Ula | Broadcast the UINFO DLPDU from station B, in the slot reserved by the UDATA_CTS. |
| | 7 | verify | DLS | Data in UINFO_a (s:= add_B) passed to DLS user | Ula | Verify that the data in the UDATA DLPDU is passed to the DLS user. |
| | 8 | verify | RF | No ACK transmitted by station A | | Verify that the station under test does not generate an ACK in response to the UDATA DLPDU. |
| postamble | 9 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Long_T_NoAck_A | | | | |
|-----------------|------|---|-----|---|-----|--|
| Purpose: | | To demonstrate that a station sends an ACK and correctly sets the T bit when in receipt of an RTS for a data packet that has already been received. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| test body | 3 | send | RF | INFO_RTS_a (s:= add_B; d:= add_A; T:= 1; pr:= 3; lg:= 7) | IRa | Send an INFO_RTS using the long transmission procedures from a simulated station B, with T = 1, indicating this is the first transmission in the sequence from station B to station A following the initialization. The RTS contains a unicast reservation for a response. |
| | 4 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 5 | verify | RF | INFO_CTS_a (s = add_A; d = add_B) in the slot reserved by the RTS | ICa | Verify that a CTS is transmitted by station A in the slot reserved by the RTS. The CTS contains an information transfer reservation that also includes a reservation for an acknowledgement. |
| | 6 | send | RF | INFO_b (s:= add_B; d:= add_A; T:= 1) in the slot reserved by the CTS | IFb | Send the DATA DLPDU from station B with T = 1 in the slot reserved by the CTS. |
| | 7 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 8 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) in the ack slot reserved by the CTS | IAa | Verify that an ACK is transmitted by station A in the ack slot reserved by the CTS. |
| | 9 | verify | RF | T = 1 in INFO_ACK_a (s = add_A; d = add_B) | IAa | Verify that the ACK contains T = 1. |
| | 10 | send | RF | INFO_RTS_a (s:= add_B; d:= add_A; T:= 1; pr:= 3; lg:= 7) | IRa | Send an RTS from station B with T = 1, to simulate the effect of the ACK not having been received by station B. The RTS contains a unicast reservation for a response. |
| | 11 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 12 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) in the slot reserved by the RTS | IAa | Verify that an ACK is transmitted by station A in the slot reserved by the RTS. |
| | 13 | verify | RF | T = 1 in INFO_ACK_a (s = add_A; d = add_B) | IAa | Verify that the ACK contains T = 1. |
| postamble | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Long_Busy_A | | | | |
|-----------------|------|---|-----|---|----------|--|
| Purpose: | | To demonstrate that a station sends either a general confirm with a unicast reservation, or a general failure, when in receipt of an RTS at a time that the channel is too busy to transmit the requested data. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the air-ground link. |
| | 4 | send | RF | INFO_RTS_a (s:= add_B; d:= add_A; T:= 1; pr:= 3; lg:= 7 (INFO length); ro:= 30; lg:= 1 (unicast length)) | IRa | Send an INFO_RTS using the long transmission procedures from a simulated station B, with T = 1, indicating this is the first transmission in the sequence from station B to station A following the initialization. The RTS contains a unicast reservation for a response. |
| | 5 | record | RF | timestartT:= time at beginning of slot in which INFO_RTS_a transmitted | | Record the time that the INFO_RTS was transmitted. |
| | 6 | rep 13 | | n:= 1 | | Start loop. |
| | 7 | send | RF | In slot beginning at time:= timestart + n: UNI_BURST_a (s:= add_C; d:= add_D; sdf:= 1; ro:= 30 + 30 * (n - 1); lg:= 15; pr:= 13) | Ua | Send a unicast burst from a simulated station C to a simulated station D, reserving successive blocks of 15 slots after the slot reserved by the unicast reservation in the RTS. |
| | 8 | send | RF | In slot beginning at time:= timestart + 1+ n: UNI_BURST_a (s:= add_E; d:= add_F; sdf:= 1; ro:= 45 + 30 * (n - 1); lg:= 15; pr:= 13) | Ua | Send a unicast burst from a simulated station E to a simulated station F, reserving successive blocks of 15 slots after the slot reserved by the unicast reservation in the RTS. |
| | 9 | endrep | | n:= n + 1 | | End loop. |
| | 10 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 11 | verify | RF | In the slot reserved by the RTS: GEN_RESP_b (s = add_A; d = add_B; r-mi = 00001101; ok = 1; err = 01 hex; sdf = 1; lg = 0; pr = 3) OR GEN_RESP_a (s = add_A; d = add_B; r-mi = 00001101; ok = 0; err = 01 hex) | GRb, GRa | Verify that station A transmits, in the slot reserved by the RTS, either a general confirm with a unicast reservation with sdf = 1, indicating when it will transmit a CTS in the future, or a general failure, with error type set to 01 hex, due to there being no slots available for transmission of the data. |
| postamble | 12 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | DLS_Long_NoInfo_A | | | | |
|-----------------|------|--|-----|---|-----|---|
| Purpose: | | To demonstrate that a station will not transmit in response to a CTS when it has no information to transmit. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the air-ground link. |
| test body | 4 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s:= add_A; d := add_B) with length = 4 slots | | Send a request from a DLS user to the station under test to send a DATA DLPDU requiring a length of 4 slots. |
| | 5 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 6 | verify | RF | INFO_RTS_a (s = add_A; d = add_B) | IRa | Verify that an INFO_RTS is sent by the station under test. The RTS contains a unicast reservation for a response. |
| | 7 | verify | RF | T = 1 in INFO_RTS_a (s = add_A; d = add_B) | IRa | Verify that T = 1 in the RTS, as this is the first transmission in the sequence from station A to station B following the initialization. |
| | 8 | send | DLS | REQUEST TO NOT TRANSMIT DATA DLPDU (s:= add_A; d := add_B) with length = 4 slots | | Send a request from a DLS user to the station under test to not transmit the DATA DLPDU, as instructed by the previous command. |
| | 9 | send | RF | INFO_CTS_a (s:= add_B; d:= add_A) in the slot reserved by the RTS | ICa | Send a CTS in response from station B to station A in the slot reserved by the RTS. |
| | 10 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 11 | verify | RF | No DATA DLPDU transmitted by station A in the slot reserved by the CTS | | Verify that station A does not transmit a DATA DLPDU in the slot reserved by the CTS. |
| postamble | 12 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Long_NACK_A | | | | | | |
|--|------|--------|-----|---|-----|--|
| Purpose: To demonstrate that a station sends a NACK when an expected data DLPDU is not received. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND3:= 5 slots; ND2:= 86 octets) | | Set the parameter ND3 = 5 slots and ND2 = 86 octets (3 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the air-ground link. |
| test body | 4 | send | RF | INFO_RTS_a (s:= add_B; d:= add_A; T:= 1; pr:= 3; lg:= 7) | IRa | Send an INFO_RTS using the long transmission procedures from a simulated station B, with T = 1, indicating this is the first transmission in the sequence from station B to station A following the initialization. The RTS contains a unicast reservation for a response. |
| | 5 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 6 | verify | RF | INFO_CTS_a (s = add_A; d = add_B) in the slot reserved by the RTS | ICa | Verify that a CTS is transmitted by station A in the slot reserved by the RTS. |
| | 7 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 8 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) in the ack slot reserved by the CTS | IAa | Verify that an ACK is transmitted by station A in the ack slot reserved by the CTS. |
| | 9 | verify | RF | T = 0 in INFO_ACK_a (s = add_A; d = add_B) | IAa | Verify that the ACK contains T = 0, indicating a negative acknowledgement, or NACK. |
| postamble | 10 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Long_NoLink | | | | | | |
|---|------|--------|-----|---|-----|--|
| Purpose: To demonstrate that a station in receipt of a CTRL_RTS with IB = 0 transmits a DM/FRMR when it does not have a link with the sender. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | CTRL_RTS_a (s:= add_B; d:= add_A; IB:= 0; T:= 0; pr:= 3; lg:= 7) | CRa | Send an RTS using the long transmission procedures from a simulated station B, with IB = 0 and T = 0. The RTS contains a unicast reservation for a response. |
| | 3 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 4 | verify | RF | DM_FRMR_a (s:= add_A; d:= add_B) in the slot reserved by the RTS | DFa | Verify that a DM/FRMR is transmitted by station A in the slot reserved by the RTS. |
| postamble | 5 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Short_NoLink | | | | | | |
|---|------|--------|-----|---|-----|---|
| Purpose: To demonstrate that a station in receipt of a CTRL transmits a DM/DISC when it does not have a link with the sender. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | CTRL_a (s:= add_B; d:= add_A; T:= 0) | CTa | Send a CTRL using the short transmission procedures from a simulated station B, with T = 0. The CTRL contains a unicast reservation for a response. |
| | 3 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 4 | verify | RF | DM_DISC_a (s:= add_A; d:= add_B) in the slot reserved by the CTRL | DDa | Verify that a DM/DISC is transmitted by station A in the slot reserved by the CTRL. |
| postamble | 5 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Long_IB_Error | | | | | | |
|--|------|--------|-----|--|-----|--|
| Purpose: To demonstrate that a station in receipt of a CTRL_RTS with IB = 1 and T = 1 transmits a DM/FRMR. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | CTRL_RTS_a (s:= add_B; d:= add_A; IB:= 1; T:= 1; pr:= 3; lg:= 7) | CRa | Send an RTS using the long transmission procedures from a simulated station B, with IB = 1 and T = 1. The RTS contains a unicast reservation for a response. |
| | 3 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 4 | verify | RF | DM_FRMR_a (s:= add_A; d:= add_B) in the slot reserved by the RTS | DFa | Verify that a DM/FRMR is transmitted by station A in the slot reserved by the RTS. |
| postamble | 5 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Long_SZOM_Error | | | | | | |
|--|------|--------|-----|--|-------|--|
| Purpose: To demonstrate that a station in receipt of an SZOM from a station with which it will only communicate using NSCOP will transmit a DM/FRMR. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | INFO_RTS_SZOM_a (s:= add_B; d:= add_A; T:= 0; pr:= 3; lg:= 7) | IRSZa | Send an INFO_RTS / SZOM using the long transmission procedures from a simulated station B, with T = 0, indicating this is the first transmission from station B to A. The burst contains a unicast reservation for a response. |
| | 3 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 4 | verify | RF | DM_FRMR_a (s:= add_A; d:= add_B) in the slot reserved by the INFO_RTS_SZOM | DFa | Verify that a DM/FRMR is transmitted by station A in the slot reserved by the INFO_RTS / SZOM. |
| postamble | 5 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_Short_SZOM_Error | | | | | | |
|--|------|--------|-----|--|-------|---|
| Purpose: To demonstrate that a station in receipt of an SZOM from a station with which it will only communicate using NSCOP will transmit a DM/FRMR. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | INFO_SZOM_a (s:= add_B; d:= add_A; T:= 0) | IFSZa | Send an INFO / SZOM using the short transmission procedures from a simulated station B, with T = 0, indicating this is the first transmission from station B to A. The burst contains a unicast reservation for a response. |
| | 3 | await | | BURST transmitted by station A | | Wait for a burst transmitted by station A. |
| | 4 | verify | RF | DM_FRMR_a (s:= add_A; d:= add_B) in the slot reserved by the INFO_SZOM | DFa | Verify that a DM/FRMR is transmitted by station A in the slot reserved by the INFO_SZOM. |
| postamble | 5 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: DLS_ND1_Short_Receive | | | | | | |
|---|------|--------|-----|--|-----|---|
| Purpose: To demonstrate that a station in receipt of a data packet from another station that is greater in length than ND1 will discard the packet. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND1:= 150 octets; ND2:= 209 octets) | | Set the parameter ND1 = 150 octets and ND2 = 209 octets (7 slots). |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| test body | 4 | send | RF | INFO_a (s = add_B; d = add_A; T = 1; in:= 163 octets of zeros) | IFa | Send a DATA DLPDU by the short transmission procedure to the station under test with T = 1 and a length of 6 slots. The INFO contains a unicast reservation for a response. |
| | 5 | verify | DLS | INFO_a has been discarded by station under test | IFa | Verify that the corrupted message has been discarded by the station under test. |
| | 6 | await | RF | BURST transmitted by station A | | Wait for a burst to be transmitted by station A. |
| | 7 | verify | RF | INFO_ACK_a (s = add_A; d = add_B) in the slot reserved by the INFO | IAa | Verify that the station under test sends an ACK in response in the slot reserved by the INFO. |
| | 8 | verify | RF | T = 1 in INFO_ACK | IAa | Verify that T = 1 in the ACK. |
| postamble | 9 | do | | SET PARAMETERS (ND1:= 1 511 octets; ND2:= 86 octets) | | Set the parameters to their default values. |
| Comments: | | | | | | |

| Test case name: | | DLS_ND1_Short_Send | | | | |
|-----------------|------|---|-----|---|-----|--|
| Purpose: | | To demonstrate that a station requested to send a data packet by a DLS user that is greater in length than ND1 will discard the packet. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | SET PARAMETERS (ND1:= 150 octets) | | Set the parameter ND1 = 150 octets. |
| | 3 | do | | M_INITIALIZE_NSCOP | | Initialize the ground-air link. |
| test body | 4 | send | DLS | REQUEST TO TRANSMIT DATA DLPDU (s = add_A; d = add_B; T = 1; in:= 163 octets of zeros) | | Send a request from a DLS user to the station under test to send a DATA DLPDU to a simulated station B requiring a burst of length 6 slots. |
| | 5 | verify | RF | No BURST transmitted by station A | | Verify that no burst has been transmitted by the station under test and hence that the corrupted message has been discarded by the station under test. |
| postamble | 6 | do | | SET PARAMETERS (ND1:= 1 511 octets) | | Set the parameter ND1 to its default value. |
| Comments: | | | | | | |

| Test case name: | | LME_LE_Station_Choose | | | | |
|-----------------|------|--|----------|---|-------|--|
| Purpose: | | To demonstrate that a mobile station will choose an appropriate ground station to establish a link, and will then establish a link with that ground station by sending a CTRL_CMD_LE (re = 1). | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | Position | Input position of station under test as laT:= 0; lon:= E 21 NM | | Inform the station under test of its own position. |
| | 3 | send | App_In | Input ground track of station under test as 90 degrees (due East) | | Inform the station under test of its own ground track. |
| | 4 | send | RF | SYNC_BURST_a (s:=add_G; laT:= CPR_LAT(0); lon:= CPR_LON(E 25 NM)) | Sa | Send a sync burst from a simulated ground station G with position indicating that it is close to the station under test and along its intended path. |
| | 5 | send | RF | SYNC_BURST_a (s:=add_H; laT:= CPR_LAT(0); lon:= CPR_LON(E 17 NM)) | Sa | Send a sync burst from a simulated ground station H with position indicating that it is close to the station under test but in the opposite direction to that in which the aircraft is heading. |
| | 6 | send | RF | SYNC_BURST_a (s:=add_I; laT:= CPR_LAT(0); lon:= CPR_LON(E 2 NM)) | Sa | Send a sync burst from a simulated ground station I with position indicating that it is far from the station under test and in the opposite direction to that in which the aircraft is heading. |
| | 7 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 8 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_G; IB:= 1; T:= 0; lg:= 1) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to simulated ground station G, with IB = 1 and T = 0, indicating this is the first transmission from station A to station G. The RTS contains a unicast reservation for a response. |
| | 9 | send | RF | CTRL_CTS_a (s = add_G; d = add_A; T = 0) in the slot reserved by the RTS | CCa | Send a CTS to the station under test from simulated ground station G. |
| | 10 | verify | RF | CTRL_d (s:= add_A; d:= add_G; M:= 0; T:= 0; re:= 1; c/r:= 0; h:= 0; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Verify that the station under test sends a CTRL DLPDU to simulated ground station G with parameters indicating a CTRL_CMD_LE (re = 1). |
| | 11 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_G; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS to the station under test. |
| | 12 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 13 | verify | RF | CTRL_CTS_a (s = add_A; d = add_G) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to simulated ground station G. |
| | 14 | send | RF | CTRL_m (s:= add_G; d:= add_A; M:= 0; T:= 0; re:= 1; c/r:= 1; h:= 0; r:= 0; lg:= 1; rrr:= 0; sss:= 0; g:= add_G) in the slot reserved by the CTS | CTm | Send a CTRL burst indicating a CTRL_RSP_LE to the station under test, including the ground-based system mask parameter. |
| | 15 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 16 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_G; T:= 0) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK in response to the completed message. |
| postamble | 17 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_LE_Ground_Invalid | | | | |
|-----------------|------|--|-----|---|-------|--|
| Purpose: | | To demonstrate that if the parameters in the CTRL_RSP_LE from the ground LME are not acceptable to the mobile LME, then the mobile LME shall transmit a DM/DISC to the ground. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | LME | REQUEST TO PERFORM LINK ESTABLISHMENT with station G | | Send a request at the LME of the station under test to perform link establishment with the selected simulated ground station G. |
| | 3 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 4 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_G; IB:= 1; T:= 0; lg:= 1) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to simulated ground station G, with IB = 1 and T = 0, indicating this is the first transmission from station A to station G. The RTS contains a unicast reservation for a response. |
| | 5 | send | RF | CTRL_CTS_a (s = add_G; d = add_A) in the slot reserved by the RTS | CCa | Send a CTS to the station under test from simulated ground station G. |
| | 6 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 7 | verify | RF | CTRL_d (s:= add_A; d:= add_G; M:= 0; T:= 0; re:= 1; c/r:= 0; h:= 0; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Verify that the station under test sends a CTRL_DLPDU to simulated ground station G with parameters indicating a CTRL_CMD_LE (re = 1). |
| | 8 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_G; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS to the station under test. |
| | 9 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 10 | verify | RF | CTRL_CTS_a (s = add_A; d = add_G) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to simulated ground station G. |
| | 11 | send | RF | CTRL_n (s:= add_G; d:= add_A; M:= 0; T:= 0; re:= 1; c/r:= 1; h:= 0; r:= 0; lg:= 1; rrr:= 0; sss:= 0; g:= add_G) in the slot reserved by the CTS | CTn | Send an invalid CTRL burst indicating a CTRL_RSP_LE from the simulated ground station G. The burst defined has a "1" in place of a "0" in the ninth octet. The burst includes the ground-based system mask parameter. |
| | 12 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 13 | verify | RF | DM_DISC_a (s:= add_A; d:= add_G) in the ack slot reserved by the CTS | DDa | Verify that a DM/DISC is transmitted by the station under test. |
| postamble | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_LE_Autotune | | | | |
|-----------------|------|---|-----|---|-------|--|
| Purpose: | | To demonstrate that if the autotune parameter is included in the CTRL_RSP_LE and the mobile LME is unable to perform the autotune, then the mobile LME shall respond with an CTRL_CMD_LCR (re = 0). | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | LME | REQUEST TO PERFORM LINK ESTABLISHMENT with station G | | Send a request at the LME of the station under test to perform link establishment with the selected simulated ground station G. |
| | 3 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 4 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_G; IB:= 1; T:= 0; lg:= 1) | CRA | Verify that the station under test sends an RTS using the long transmission procedures to ground station G, with IB = 1 and T = 0, indicating this is the first transmission from station A to station G. The RTS contains a unicast reservation for a response. |
| | 5 | send | RF | CTRL_CTS_a (s = add_G; d = add_A) in the slot reserved by the RTS | CCa | Send a CTS to the station under test from ground station G. |
| | 6 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 7 | verify | RF | CTRL_d (s:= add_A; d:= add_G; M:= 0; T:= 0; re:= 1; c/r:= 0; h:= 0; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Verify that the station under test sends a CTRL DLPDU to ground station G with parameters indicating a CTRL_CMD_LE (re = 1). |
| | 8 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_G; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS to the station under test. |
| | 9 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 10 | verify | RF | CTRL_CTS_a (s = add_A; d = add_G) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to ground station G. |
| | 11 | send | RF | CTRL_o (s:= add_G; d:= add_A; re:= 1; c/r:= 1; h:= 0; r:= 0; lg:= 1; rrr:= 0; sss:= 0; m:= 6; f:= 000; lg:=4; g:= add_G; g:= add_G) in the slot reserved by the CTS | CTo | Send a CTRL burst indicating a CTRL_RSP_LE from the simulated ground station G, which the mobile cannot support as the modulation encoding is unrecognized. The burst includes the autotune, replacement ground station list, and ground-based system mask parameters. |
| | 12 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 13 | verify | RF | CTRL_ACK_CTRL_RTS_a (s = add_A; d = add_G; lg:= 2) in the ack slot reserved by the CTS | CACRa | Verify that the station under test sends a CTRL_ACK / CTRL_RTS to ground station G. |
| | 14 | send | RF | CTRL_CTS_a (s = add_G; d = add_A) in the slot reserved by the RTS | CCa | Send a CTS to the station under test from ground station G. |
| | 15 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 16 | verify | RF | CTRL_s (s:= add_A; d:= add_G; M:= 0; re:= 0; c/r:= 0; h:= 0; r:= 1; lg:= 1; rrr:= 0; sss:= 0; lg:= 6; c:= 05hex; d:= 0; a:= 0) in the slot reserved by the CTS | CTs | Verify that a combined CTRL burst indicating a CTRL_CMD_LCR (re = 0) is sent by the station under test. The burst includes the LCR cause parameter. |
| | 17 | send | RF | CTRL_ACK (s = add_G; d = add_A) in the ack slot reserved by the CTS | CAa | Send a CTRL_ACK to the station under test. |
| postamble | 18 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_Mob_Init | | | | |
|-----------------|------|---|-----|---|-------|--|
| Purpose: | | To demonstrate that a mobile that wishes to initiate a handoff to a new ground station will send to that ground station a CTRL_CMD_HO (re = 1) with parameters correctly defined. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | LME | REQUEST TO TRANSMIT MOBILE-INITIATED HANDOFF to new ground station H | | Send a request at the LME of the station under test to transmit a mobile-initiated handoff to simulated new ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; lB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to new ground station H. The RTS contains a unicast reservation for a response. |
| | 6 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test. |
| | 7 | verify | RF | CTRL_q (s:= add_A; d:= add_H; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS | CTq | Verify that the station under test sends a CTRL DLPDU to station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options parameter. |
| | 8 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_H; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS from station H. |
| | 9 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 10 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to station H. |
| | 11 | send | RF | CTRL_d (s:= add_H; d:= add_A; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Send a CTRL burst indicating a CTRL_RSP_HO from station H. |
| | 12 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK in response to the completed message. |
| postamble | 13 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_Mob_Init_Auto | | | | |
|-----------------|------|---|-----|--|-------|---|
| Purpose: | | To demonstrate that in the case of a mobile-initiated handoff in which the ground station has included the autotune parameter in the CTRL_RSP_HO, and where the mobile cannot support the autotune, the mobile will transmit a CTRL_CMD_LCR (re = 0). | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | LME | REQUEST TO TRANSMIT MOBILE-INITIATED HANDOFF to new ground station H | | Send a request at the LME of the station under test to transmit a mobile-initiated handoff to simulated new ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; lB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to new ground station H. The RTS contains a unicast reservation for a response. |
| | 6 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test. |
| | 7 | verify | RF | CTRL_q (s:= add_A; d:= add_H; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS | CTq | Verify that the station under test sends a CTRL DLPDU to station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options parameter. |
| | 8 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_H; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS from station H. |
| | 9 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 10 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to station H. |
| | 11 | send | RF | CTRL_k (s:= add_H; d:= add_A; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0; m:= 4; f (bit 12):= 1; f (bits 1 to 11):= 000hex; lg:=4; g:= add_l) in the slot reserved by the CTS | CTk | Send a CTRL burst indicating a CTRL_RSP_HO from station H, including the autotune and replacement ground station list parameters. The mobile is unable to support the frequency listed as bit 12 = 1 indicates that it is reserved for future allocation. |
| | 12 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 13 | verify | RF | CTRL_ACK_CTRL_RTS_a (s = add_A; d = add_H; lg:= 2) in the ack slot reserved by the CTS | CACRa | Verify that the station under test sends a CTRL_ACK / CTRL_RTS to station H. |
| | 14 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTS to the station under test from station H. |
| | 15 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 16 | verify | RF | CTRL_s (s:= add_A; d:= add_H; M:= 0; re:= 0; c/r:= 0; h:= 0; r:= 1; lg:= 1; rrr:= 0; sss:= 0; lg:= 6; c:= 05hex; d:= 0; a:= 0) in the slot reserved by the CTS | CTs | Verify that a combined CTRL burst indicating a CTRL_CMD_LCR (re = 0) is sent by the station under test. The burst includes the LCR cause parameter. |
| | 17 | send | RF | CTRL_ACK (s = add_H; d = add_A) in the ack slot reserved by the CTS | CAa | Send a CTRL_ACK to the station under test. |
| postamble | 18 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_Mob_Init_Grd_Invalid | | | | |
|-----------------|------|--|-----|---|-------|--|
| Purpose: | | To demonstrate that in the case of a mobile-initiated handoff in which the parameters of the CTRL_RSP_HO transmitted by the ground station are not acceptable to the mobile, the mobile will transmit a DM/DISC to the ground on the new link. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | LME | REQUEST TO TRANSMIT MOBILE-INITIATED HANDOFF to new ground station H | | Send a request at the LME of the station under test to transmit a mobile-initiated handoff to simulated new ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to new ground station H. The RTS contains a unicast reservation for a response. |
| | 6 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test. |
| | 7 | verify | RF | CTRL_q (s:= add_A; d:= add_H; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS | CTq | Verify that the station under test sends a CTRL DLPDU to station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options parameter. |
| | 8 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_H; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS from station H. |
| | 9 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 10 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to station H. |
| | 11 | send | RF | CTRL_e (s:= add_H; d:= add_A; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTe | Send an invalid CTRL burst indicating a CTRL_RSP_HO from station H The burst defined has a "1" in place of a "0" in the ninth octet. |
| | 12 | verify | RF | DM_DISC_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | DDa | Verify that a DM/DISC is transmitted by the station under test. |
| postamble | 13 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_TL2 | | | | |
|-----------------|------|---|-----|---|-------|--|
| Purpose: | | To demonstrate that a mobile LME will establish a link to a new ground station if TL2 seconds have elapsed since the LME initiated the request to send any burst to the current ground station. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | LME | REQUEST TO TRANSMIT MOBILE-REQUESTED GROUND-INITIATED HANDOFF addressed to current ground station G | | Send a request at the LME of the station under test to transmit a mobile-requested ground-initiated handoff addressed to current ground station G. |
| | 4 | record | LME | tl2_time:= time at which request initiated at LME | | Record the time that the request was initiated at the LME. |
| | 5 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 6 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_G; IB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to simulated current ground station G. The RTS contains a unicast reservation for a response. |
| | 7 | wait | | TL2 seconds | | Wait for TL2 seconds, after which time the mobile LME should attempt to handoff to another ground station. |
| | 8 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 9 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to simulated new ground station H. The RTS contains a unicast reservation for a response. |
| | 10 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS from station H to the station under test. |
| | 11 | verify | RF | CTRL_q (s:= add_A; d:= add_H; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS | CTq | Verify that the station under test sends a CTRL DLPDU to station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options parameter. |
| | 12 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_H; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS from station H. |
| | 13 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 14 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to station H. |
| | 15 | send | RF | CTRL_d (s:= add_H; d:= add_A; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Send a CTRL burst indicating a CTRL_RSP_HO from station H. |
| | 16 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK in response to the completed message. |
| postamble | 17 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_TM2 | | | | |
|-----------------|------|---|-----------|--|--------|--|
| Purpose: | | To demonstrate that a mobile LME will autonomously tune to an alternate frequency (provided in a frequency support list) and then establish a link to a new ground station, if timer TM2 expires. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | LME | REQUEST TO PERFORM LINK ESTABLISHMENT with ground station G | | Send a request at the LME of the station under test to perform link establishment with the selected simulated ground station G. |
| | 3 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 4 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_G; IB:= 1; T:= 0; lg:= 1) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to simulated ground station G, with IB = 1 and T = 0, indicating this is the first transmission from station A to station G. The RTS contains a unicast reservation for a response. |
| | 5 | send | RF | CTRL_CTS_a (s = add_G; d = add_A; T = 0) in the slot reserved by the RTS | CCa | Send a CTS to the station under test from simulated ground station G. |
| | 6 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 7 | verify | RF | CTRL_d (s:= add_A; d:= add_G; M:= 0; T:= 0; re:= 1; c/r:= 0; h:= 0; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Verify that the station under test sends a CTRL DLPDU to simulated ground station G with parameters indicating a CTRL_CMD_LE (re = 1). |
| | 8 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_G; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS to the station under test. |
| | 9 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 10 | verify | RF | CTRL_CTS_a (s = add_A; d = add_G) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to simulated ground station G. |
| | 11 | send | RF | CTRL_m (s:= add_G; d:= add_A; M:= 0; T:= 0; re:= 1; c/r:= 1; h:= 0; r:= 0; lg:= 1; rrr:= 0; sss:= 0; g:= add_G; p, VS3:= default values; TM2:= 25 slots) in the slot reserved by the CTS | CTt | Send a CTRL burst indicating a CTRL_RSP_LE to the station under test, including the ground-based system mask and random access parameters. The random access parameter sets TM2 to 25 slots. |
| | 12 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 13 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_G; T:= 0) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK in response to the completed message. |
| | 14 | send | RF (GSC1) | SYNC_BURST_d(16) (s = add_C) | Sd(16) | Send a sync burst from a simulated station C extending over 16 slots. |
| | 15 | record | RF (GSC1) | sync_time:= time at beginning of slot containing SYNC_BURST_d(16) | Sd(16) | Record the time the sync burst was transmitted. |
| | 16 | await | | time:= sync_time + 16 × 60/M1 | | |
| | 17 | send | RF (GSC1) | SYNC_BURST_d(16) (pT:= 3; po:= 0; s= add_D) in slot beginning attime:= sync_time + 16 × 60/M1 | Sd(16) | Send a sync burst from a simulated station D extending over 16 slots. |
| | 18 | send | LME | GSIF TO SET PARAMETERS (s:= add_G; TM2:= 25 slots) | | Send a GSIF from ground station G to set the channel busy timer TM2 to expire after 25 slots. |
| | 19 | await | | time:= sync_time + 60 + 1 × 60/M1 | | |
| | 20 | send | LME | REQUEST TO TRANSMIT MOBILE-REQUESTED GROUND-INITIATED HANDOFF addressed to current ground station G at time:= sync_time + 60 + 1 × 60/M1 | | Send a request at the LME of the station under test to transmit a mobile-requested ground-initiated handoff addressed to current ground station G. |

| Test case name: | | LME_HO_TM2 | | | | |
|-----------------|------|---|-----------|---|-------|--|
| Purpose: | | To demonstrate that a mobile LME will autonomously tune to an alternate frequency (provided in a frequency support list) and then establish a link to a new ground station, if timer TM2 expires. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 21 | verify | LME | message sent to LME notifying congestion | | Verify at LME that congestion is notified. |
| | 22 | await | RF (GSC2) | BURST from station under test | | Wait for a burst from the station under test on the only other alternate frequency available GSC2. |
| | 23 | verify | RF (GSC2) | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to simulated new ground station H on the GSC2 frequency. The RTS contains a unicast reservation for a response. |
| | 24 | send | RF (GSC2) | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS from station H to the station under test. |
| | 25 | verify | RF (GSC2) | CTRL_q (s:= add_A; d:= add_H; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS | CTq | Verify that the station under test sends a CTRL DLPDU to station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options parameter. |
| | 26 | send | RF (GSC2) | CTRL_ACK_CTRL_RTS_a (s = add_H; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS from station H. |
| | 27 | await | RF (GSC2) | BURST from station under test | | Wait for a burst from the station under test. |
| | 28 | verify | RF (GSC2) | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to station H. |
| | 29 | send | RF (GSC2) | CTRL_d (s:= add_H; d:= add_A; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Send a CTRL burst indicating a CTRL_RSP_HO from station H. |
| | 30 | verify | RF (GSC2) | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK in response to the completed message. |
| postamble | 31 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: LME_HO_Mob_Req_Grd_Init_A | | | | | | |
|---|------|--------|-----|--|-------|--|
| Purpose: To demonstrate that a mobile that wishes to request the ground LME to initiate a handoff, addressed to its current ground station, will send a CTRL_CMD_HO (re = 0) to that station. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | LME | REQUEST TO TRANSMIT MOBILE-REQUESTED GROUND-INITIATED HANDOFF addressed to current ground station G | | Send a request at the LME of the station under test to transmit a mobile-requested ground-initiated handoff addressed to current ground station G. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_c (s:= add_A; d:= add_G; M:= 0; T:= 1; re:= 0; c/r:= 0; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) | CTc | Verify that the station under test sends a CTRL DLPDU to current ground station G with parameters indicating a CTRL_CMD_HO (re = 0). The CTRL contains a unicast reservation for a response. |
| | 6 | send | RF | CTRL_ACK_a (s:= add_G; d:= add_A) in the ack slot reserved by the CTRL | CAa | Send the ACK from current station G in response. |
| | 7 | send | RF | CTRL_RTS_a (s = add_H; d = add_A; lg:= 2) | CRa | Send a CTRL_RTS to the station under test from proposed new ground station H. |
| | 8 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 9 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to new ground station H. |
| | 10 | send | RF | CTRL_h (s:= add_H; d:= add_A; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; l:= 1; lg:= 6; a:= ADM; r:= ARS; g:= add_H) in the slot reserved by the CTS | CTh | Send a CTRL DLPDU from new ground station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options, ATN router NETs, and ground-based system mask parameters. |
| | 11 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 12 | verify | RF | CTRL_ACK_CTRL_RTS_a (s = add_A; d = add_H; lg:= 2) in the ack slot reserved by the CTS | CACRa | Verify that a CTRL_ACK / CTRL_RTS is sent by the station under test to the new ground station H. |
| | 13 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test from new ground station H. |
| | 14 | verify | RF | CTRL_d (s:= add_A; d:= add_H; M:= 0; T:= 0; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Verify that a CTRL burst indicating a CTRL_RSP_HO is sent by the station under test to new ground station H. |
| | 15 | send | RF | CTRL_ACK_a (s:= add_H; d:= add_A) in the ack slot reserved by the CTS | CAa | Send the ACK from new station H in response to the completed message. |
| postamble | 16 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: LME_HO_Mob_Req_Grd_Init_B | | | | | | |
|--|------|--------|-----|--|-------|--|
| Purpose: To demonstrate that a mobile that wishes to request the ground LME to initiate a handoff, addressed to its proposed ground station, will send a CTRL_CMD_HO (re = 0) to that station. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | LME | REQUEST TO TRANSMIT MOBILE-REQUESTED GROUND-INITIATED HANDOFF addressed to proposed ground station H | | Send a request at the LME of the station under test to transmit a mobile-requested ground-initiated handoff addressed to proposed ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_RTS_a (s = add_A; d = add_H; lg = 1) in the ack slot reserved by the CTS | CRa | Verify that a CTRL_ACK / CTRL_RTS is sent by the station under test to the new ground station H. |
| | 6 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test from new ground station H. |
| | 7 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 8 | verify | RF | CTRL_d (s := add_A; d := add_H; M := 0; T := 1; re := 0; c/r := 0; h := 1; r := 0; lg := 1; rrr := 0; sss := 0) | CTd | Verify that the station under test sends a CTRL DLPDU to proposed new ground station H with parameters indicating a CTRL_CMD_HO (re = 0). |
| | 9 | send | RF | CTRL_ACK_CTRL_RTS_a (s := add_H; d := add_A; lg := 2) in the ack slot reserved by the CTRL | CACRa | Send a CTRL_ACK / CTRL_RTS from new ground station H in response. |
| | 10 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 11 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to new ground station H. |
| | 12 | send | RF | CTRL_h (s := add_H; d := add_A; re := 1; c/r := 0; lg := 1; h := 1; r := 0; rrr := 0; sss := 0; b := 0; l := 1; lg := 6; a := ADM; r := ARS; g := add_H) in the slot reserved by the CTS | CTh | Send a CTRL DLPDU from new ground station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options, ATN router NETs, and ground-based system mask parameters. |
| | 13 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 14 | verify | RF | CTRL_ACK_CTRL_RTS_a (s = add_A; d = add_H; lg := 2) in the ack slot reserved by the CTS | CACRa | Verify that a CTRL_ACK / CTRL_RTS is sent by the station under test to the new ground station H. |
| | 15 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test from new ground station H. |
| | 16 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 17 | verify | RF | CTRL_d (s := add_A; d := add_H; M := 0; T := 0; re := 1; c/r := 1; h := 1; r := 0; lg := 1; rrr := 0; sss := 0) in the slot reserved by the CTS | CTd | Verify that a CTRL burst indicating a CTRL_RSP_HO is sent by the station under test to new ground station H. |
| | 18 | send | RF | CTRL_ACK_a (s := add_H; d := add_A) in the ack slot reserved by the CTS | CAa | Send the ACK from new station H in response to the completed message. |
| postamble | 19 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_Mob_Req_Grd_Init_Aff | | | | |
|-----------------|------|--|-----|--|-----|--|
| Purpose: | | To demonstrate that in the case of a mobile-requested ground-initiated handoff, the current link remains unaffected after transmission of the CTRL_CMD_HO (re = 0) while the mobile LME has not yet received a CTRL_CMD_HO (re = 1). | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | LME | REQUEST TO TRANSMIT MOBILE-REQUESTED GROUND-INITIATED HANDOFF addressed to current ground station G | | Send a request at the LME of the station under test to transmit a mobile-requested ground-initiated handoff addressed to current ground station G. |
| | 4 | record | LME | timestarT:= time at which request initiated at LME | | Record the time that the request was initiated at the LME. |
| | 5 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 6 | verify | RF | CTRL_c (s:= add_A; d:= add_G; M:= 0; T:= 1; re:= 0; c/r:= 0; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) | CTc | Verify that the station under test sends a CTRL DLPDU to current ground station G with parameters indicating a CTRL_CMD_HO (re = 0). The CTRL contains a unicast reservation for a response. |
| | 7 | send | RF | CTRL_ACK_a (s:= add_G; d:= add_A) in the slot reserved by the CTRL | CAa | Send the ACK from current ground station G in response to the completed message. |
| | 8 | rep 4 | | m:= {1, 2, 3, 4}; info:= {info1, info2, info3, info4, info5} | | Start loop to send packets on the old link before TL2 expires. All the "infox" packets require a short transmission. |
| | 9 | await | | time:= timestart + m | | Wait for a specified time after the request initiated at the LME. |
| | 10 | send | RF | INFO_a (s:= add_G; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst from the current ground station G to the station under test with M = 0. The INFO contains a unicast reservation for a response. |
| | 11 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_G) in the slot reserved by the INFO | IAa | Verify INFO_ACK received from the station under test addressed to station G, and therefore that the link remains unaffected. |
| | 12 | endrep | | | | End loop. |
| postamble | 13 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_Gr_Req_Mob_Init | | | | |
|-----------------|------|--|-----|---|-------|---|
| Purpose: | | To demonstrate that a mobile that receives a CTRL_CMD_HO (re = 0) from a ground station will perform a mobile initiated handoff. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | RF | CTRL_c (s:= add_G; d:= add_A; re:= 0; c/r:= 0; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) | CTc | Send a CTRL DLPDU to the station under test from current ground station G, with parameters indicating a CTRL_CMD_HO (re = 0). The CTRL contains a unicast reservation for a response. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_G) in the ack slot reserved by the CTRL | CAa | Verify that the station under test sends an ACK to station G in response. |
| | 6 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 7 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to new ground station H. The RTS contains a unicast reservation for a response. |
| | 8 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test. |
| | 9 | verify | RF | CTRL_q (s:= add_A; d:= add_H; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS | CTq | Verify that the station under test sends a CTRL DLPDU to station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options parameter. |
| | 10 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_H; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS from station H. |
| | 11 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 12 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to station H. |
| | 13 | send | RF | CTRL_d (s:= add_H; d:= add_A; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Send a CTRL burst indicating a CTRL_RSP_HO from station H. |
| | 14 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK in response to the completed message. |
| postamble | 15 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_Gr_Req_Mob_Init_Invalid | | | | |
|-----------------|------|---|-----|---|-------|--|
| Purpose: | | To demonstrate that a mobile that receives an invalid CTRL_CMD_HO (re = 0) from a ground station will send a CTRL_CMD_LCR (re = 0). | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | RF | CTRL_g (s:= add_G; d:= add_B; re:= 0; c/r:= 0; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) | CTg | Send an invalid CTRL DLPDU from current ground station G with parameters indicating a CTRL_CMD_HO (re = 0). The CTRL contains a unicast reservation for a response. The burst contains a "1" in place of a "0" in octet 9. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_ACK_CTRL_RTS_a (s = add_A; d = add_G; lg:= 1) in the slot reserved by the CTRL | CACRa | Verify that a CTRL_ACK / CTRL_RTS is sent by the station under test. |
| | 6 | send | RF | CTRL_CTS_a (s = add_G; d = add_A) in the slot reserved by the RTS | CCa | Send a CTS to the station under test from station G. |
| | 7 | verify | RF | CTRL_s (s:= add_A; d:= add_G; re:= 0; c/r:= 0; h:= 0; r:= 1; lg:= 1; rrr:= 0; sss:= 0; lg:= 6; c:= 81hex; d:= 0; a(first octet):= 00001000; a(second octet):= 01000001) in the slot reserved by the CTS | CTs | Verify that a CTRL burst indicating a CTRL_CMD_LCR (re = 0) is sent by the station under test. The burst includes the LCR cause parameter. |
| | 8 | send | RF | CTRL_ACK_a (s:= add_G; d:= add_A) in the ack slot reserved by the CTS | CAa | Send the ACK from station G in response to the completed message. |
| postamble | 9 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_Gr_Req_Mob_Init_Auto | | | | |
|-----------------|------|--|-----------|---|-------|--|
| Purpose: | | To demonstrate that a mobile that receives a CTRL_CMD_HO (re = 0) from a ground station that includes the autotune parameter will perform a mobile initiated handoff on the new frequency. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G on GSC1). |
| | 3 | send | RF (GSC1) | CTRL_k (s:= add_G; d:= add_A; re:= 0; c/r:= 0; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0; m:= 4; f:= GSC2; lg:=4; g:= add_H) | CTk | Send a CTRL DLPDU to the station under test from current ground station G, with parameters indicating a CTRL_CMD_HO (re = 0), and including the autotune and replacement ground station list parameters. The CTRL contains a unicast reservation for a response. |
| | 4 | await | RF (GSC1) | BURST from station under test | | Wait for a burst from the station under test (on GSC1). |
| | 5 | verify | RF (GSC1) | CTRL_ACK_a (s:= add_A; d:= add_G) in the ack slot reserved by the CTRL | CAa | Verify that the station under test sends an ACK to station G in response (on GSC1). |
| | 6 | await | RF (GSC2) | BURST from station under test | | Wait for a burst from the station under test on GSC2. |
| | 7 | verify | RF (GSC2) | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS to new ground station H on GSC2 using the long transmission procedures. The RTS contains a unicast reservation for a response. |
| | 8 | send | RF (GSC2) | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test. |
| | 9 | verify | RF (GSC2) | CTRL_q (s:= add_A; d:= add_H; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS | CTq | Verify that the station under test sends a CTRL DLPDU to station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options parameter. |
| | 10 | send | RF (GSC2) | CTRL_ACK_CTRL_RTS_a (s = add_H; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS from station H. |
| | 11 | await | RF (GSC2) | BURST from station under test | | Wait for a burst from the station under test. |
| | 12 | verify | RF (GSC2) | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to station H. |
| | 13 | send | RF (GSC2) | CTRL_d (s:= add_H; d:= add_A; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Send a CTRL burst indicating a CTRL_RSP_HO from station H. |
| | 14 | verify | RF (GSC2) | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK in response to the completed message. |
| postamble | 15 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: LME_HO_Gr_Req_Mob_Init_Auto_Retrans | | | | | | |
|---|------|--------|-----------|---|-----|--|
| Purpose: To demonstrate that a mobile that receives a CTRL_CMD_HO (re = 0) from a ground station that includes the autotune parameter will attempt to retransmit a CTRL_CMD_HO (re = 1) according to the retransmission procedures on the new frequency if it receives no response. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G on GSC1). |
| | 3 | send | RF (GSC1) | CTRL_k (s:= add_G; d:= add_A; re:= 0; c/r:= 0; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0; m:= 4; f:= GSC2; lg:=4; g:= add_H) | CTk | Send a CTRL DLPDU to the station under test from current ground station G, with parameters indicating a CTRL_CMD_HO (re = 0), and including the autotune and replacement ground station list parameters. The CTRL contains a unicast reservation for a response. |
| | 4 | await | RF (GSC1) | BURST from station under test | | Wait for a burst from the station under test (on GSC1). |
| | 5 | verify | RF (GSC1) | CTRL_ACK_a (s:= add_A; d:= add_G) in the ack slot reserved by the CTRL | CAa | Verify that the station under test sends an ACK to station G in response (on GSC1). |
| | 6 | await | RF (GSC2) | BURST from station under test | | Wait for a burst from the station under test on GSC2. |
| | 7 | verify | RF (GSC2) | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS to new ground station H on GSC2 using the long transmission procedures. The RTS contains a unicast reservation for a response. |
| | 8 | record | RF(GSC2) | timedata1:= time at beginning of slot containing CTRL_RTS_a | CRa | Record the time that the RTS was transmitted by the station under test. |
| | 9 | await | RF (GSC2) | BURST from station under test | | Wait for a burst from the station under test on GSC2. |
| | 10 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) transmitted before time:= timedata1 + 2,45 | CRa | Verify that the station under test attempts to retransmit the RTS. |
| | 11 | record | RF(GSC2) | timedata2:= time at beginning of slot containing CTRL_RTS_a | CRa | Record the time that the RTS was transmitted by the station under test. |
| | 12 | await | RF (GSC2) | BURST from station under test | | Wait for a burst from the station under test on GSC2. |
| | 13 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) transmitted before time:= timedata2 + 3,465 | CRa | Verify that the station under test attempts to retransmit the RTS. |
| | 14 | record | RF(GSC2) | timedata3:= time at beginning of slot containing CTRL_RTS_a | CRa | Record the time that the RTS was transmitted by the station under test. |
| | 15 | await | RF (GSC2) | BURST from station under test | | Wait for a burst from the station under test on GSC2. |
| | 16 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) transmitted before time:= timedata3 + 4,1905 | CRa | Verify that the station under test attempts to retransmit the RTS. |
| | 17 | record | RF(GSC2) | timedata4:= time at beginning of slot containing CTRL_RTS_a | CRa | Record the time that the RTS was transmitted by the station under test. |
| | 18 | await | RF (GSC2) | BURST from station under test | | Wait for a burst from the station under test on GSC2. |

| Test case name: LME_HO_Gr_Req_Mob_Init_Auto_Retrans | | | | | | |
|---|------|--------|-----|---|-----|---|
| Purpose: To demonstrate that a mobile that receives a CTRL_CMD_HO (re = 0) from a ground station that includes the autotune parameter will attempt to retransmit a CTRL_CMD_HO (re = 1) according to the retransmission procedures on the new frequency if it receives no response. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 19 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) transmitted before time:= timedata4 + 8,12385 | CRa | Verify that the station under test attempts to retransmit the RTS. |
| | 20 | verify | LME | notification received that Q5num attempts have been made to transmit CTRL_RTS_a on the new frequency with no response | | Verify that the station notifies the LME that Q5num attempts have been made to retransmit the RTS, and that no response was received following any of the attempts. |
| postamble | 21 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: LME_HO_Receive_Invalid | | | | | | |
|--|------|--------|-----|--|-------|--|
| Purpose: To demonstrate that a mobile in receipt of a handoff command from a ground station with which it does not have a link will transmit a CTRL_LCR. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | CTRL_RTS_a (s:= add_H; d:= add_A; IB:= 1; T:= 0; lg:= 2) | CRa | Send an RTS from ground station H. The station under test has no current link with this ground station. |
| | 3 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 4 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to ground station H. |
| | 5 | send | RF | CTRL_h (s:= add_H; d:= add_A; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; l:= 1; lg:= 6; a:= ADM; r:= ARS; g:= add_H) in the slot reserved by the CTS | CTh | Send a CTRL DLPDU from ground station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options, ATN router NETs, and ground-based system mask parameters. |
| | 6 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 7 | verify | RF | CTRL_ACK_CTRL_RTS_a (s = add_A; d = add_H; lg:= 2) in the ack slot reserved by the CTS | CACRa | Verify that a CTRL_ACK / CTRL_RTS is sent by the station under test to ground station H. |
| | 8 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test. |
| | 9 | verify | RF | CTRL_s (s:= add_A; d:= add_H; M:= 0; T:= 0; re:= 1; c/r:= 1; h:= 0; r:= 1; lg:= 1; rrr:= 0; sss:= 0 lg:= 6; c:= 81hex; d:= 0; a(first octet):= 00010110; a(second octet):= 00000001) in the slot reserved by the CTS | CTs | Verify that a CTRL burst indicating a CTRL_RSP_LCR is sent by the station under test to ground station H. |
| | 10 | send | RF | CTRL_ACK_a (s:= add_H; d:= add_A) in the ack slot reserved by the CTS | CAa | Send the ACK from station H in response to the completed message. |
| postamble | 11 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_Ground_Init | | | | |
|-----------------|------|---|-----|--|-------|--|
| Purpose: | | To demonstrate that a station displays correct operation when handoff is initiated by the ground station. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | RF | CTRL_RTS_a (s:= add_H; d:= add_A; lB:= 1; T:= 0; lg:= 2) | CRa | Send an RTS from proposed new ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to new ground station H. |
| | 6 | send | RF | CTRL_h (s:= add_H; d:= add_A; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; l:= 1; lg:= 6; a:= ADM; r:= ARS; g:= add_H) in the slot reserved by the CTS | CTh | Send a CTRL DLPDU from new ground station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options, ATN router NETs, and ground-based system mask parameters. |
| | 7 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 8 | verify | RF | CTRL_ACK_CTRL_RTS_a (s = add_A; d = add_H; lg:= 2) in the ack slot reserved by the CTS | CACRa | Verify that a CTRL_ACK / CTRL_RTS is sent by the station under test to new ground station H. |
| | 9 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test. |
| | 10 | verify | RF | CTRL_d (s:= add_A; d:= add_H; M:= 0; T:= 0; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Verify that a CTRL burst indicating a CTRL_RSP_HO is sent by the station under test to new ground station H. |
| | 11 | send | RF | CTRL_ACK_a (s:= add_H; d:= add_A) in the ack slot reserved by the CTS | CAa | Send the ACK from new station H in response to the completed message. |
| postamble | 12 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_Ground_Init_Invalid | | | | |
|-----------------|------|---|-----|--|-------|--|
| Purpose: | | To demonstrate that in the case of a ground-initiated handoff, a mobile LME which cannot accept the handoff request will transmit a CTRL_RSP_LCR. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | RF | CTRL_RTS_a (s:= add_H; d:= add_A; IB:= 1; T:= 0; lg:= 2) | CRA | Send an RTS from proposed new ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to new ground station H. |
| | 6 | send | RF | CTRL_r (s:= add_H; d:= add_A; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; l:= 1; lg:= 6; a:= ADM; r:= ARS; g:= add_H) in the slot reserved by the CTS | CTr | Send an invalid CTRL DLPDU from new ground station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options, ATN router NETs, and ground-based system mask parameters (the burst defined has a "1" in place of a "0" in the ninth octet). |
| | 7 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 8 | verify | RF | CTRL_ACK_CTRL_RTS_a (s = add_A; d = add_H; lg:= 2) in the ack slot reserved by the CTS | CACRa | Verify that a CTRL_ACK / CTRL_RTS is sent by the station under test to new ground station H. |
| | 9 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test. |
| | 10 | verify | RF | CTRL_s (s:= add_A; d:= add_H; M:= 0; T:= 0; re:= 1; c/r:= 1; h:= 0; r:= 1; lg:= 1; rrr:= 0; sss:= 0 lg:= 6; c:= 81hex; d:= 0; a(first octet):= 00001000; a(second octet):= 01000001) in the slot reserved by the CTS | CTs | Verify that a CTRL burst indicating a CTRL_RSP_LCR is sent by the station under test to new ground station H. |
| | 11 | send | RF | CTRL_ACK_a (s:= add_H; d:= add_A) in the ack slot reserved by the CTS | CAa | Send the ACK from new station H in response to the completed message. |
| Postamble | 12 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: LME_HO_Ground_Init_List | | | | | | |
|--|------|--------|----------|--|-------|---|
| Purpose: To demonstrate that a station displays correct operation when handoff is initiated by the ground station, and when the replacement ground station list parameter is provided by the ground station, and when a preferable ground station is included in the list. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | Position | Input position of station under test as laT:= 0; lon:= E 22 NM | | Inform the station under test of its own position. |
| | 4 | send | App_In | Input ground track of station under test as 90 degrees (due East) | | Inform the station under test of its own ground track. |
| | 5 | send | RF | SYNC_BURST_a (s:=add_G; laT:= CPR_LAT(0); lon:= CPR_LON(E 5 NM)) | Sa | Send a sync burst from a current ground station G with position such that the station under test is moving away from it. |
| | 6 | send | RF | SYNC_BURST_a (s:=add_H; laT:= CPR_LAT(0); lon:= CPR_LON(E 13 NM)) | Sa | Send a sync burst from a simulated ground station H with a position that is closer to the station under test but such that the station under test is moving away from it. |
| | 7 | send | RF | SYNC_BURST_a (s:=add_J; laT:= CPR_LAT(0); lon:= CPR_LON(E 25 NM)) | Sa | Send a sync burst from a simulated ground station J with position indicating that it is close to the station under test and in the direction in which the aircraft is heading. |
| | 8 | send | RF | CTRL_RTS_a (s:= add_H; d:= add_A; IB:= 1; T:= 0; lg:= 2) | CRa | Send an RTS from proposed new ground station H (this station has been proposed by the ground LME). |
| | 9 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 10 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to proposed new ground station H. |
| | 11 | send | RF | CTRL_i (s:= add_H; d:= add_A; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; l:= 1; lg:= 6; a:= ADM; r:= ARS; g:= add_H; lg:=6; g:= add_J) in the slot reserved by the CTS | CTi | Send a CTRL DLPDU from proposed new ground station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options, ATN router NETs, ground-based system mask, and replacement ground station list parameters. The replacement ground station list parameter includes a ground station J which is preferable to the mobile. |
| | 12 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 13 | verify | RF | CTRL_ACK_a (s = add_A; d = add_H) in the slot reserved by the CTS | CACRa | Verify that a CTRL_ACK is sent by the station under test to proposed new ground station H. |
| | 14 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 15 | verify | RF | CTRL_RTS_a (s = add_A; d = add_J; lg:= 2) in the slot reserved by the CTRL_ACK if applicable | CACRa | Verify that a CTRL_ACK / CTRL_RTS is sent by the station under test to its preferred new ground station J. |
| | 16 | send | RF | CTRL_CTS_a (s = add_J; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test from preferred new ground station J. |
| | 17 | verify | RF | CTRL_d (s:= add_A; d:= add_J; M:= 0; T:= 0; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Verify that a CTRL burst indicating a CTRL_RSP_HO is sent by the station under test to preferred new ground station J. |
| | 18 | send | RF | CTRL_ACK_a (s:= add_J; d:= add_A) in the ack slot reserved by the CTS | CAa | Send the ACK from preferred new ground station J in response to the completed message. |
| postamble | 19 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_HO_Ground_Broadcast | | | | |
|-----------------|------|---|-----|--|-------|--|
| Purpose: | | To demonstrate that a station displays correct operation when requested to comply with a broadcast handoff. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | RF | CTRL_RTS_a (s:= add_H; d:= add_A; lB:= 1; T:= 0; lg:= 2) | CRa | Send an RTS from proposed new ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to new ground station H. |
| | 6 | send | RF | UCTRL_b (s:= add_H; ucid:= 3; re:= 0; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; l:= 0; b:= 1; g:= add_G; lg:= 4; a:= add_A; lg:= 6; a:= ADM; r:= ARS) in the slot reserved by the CTS | UCb | Send a CTRL DLPDU from new ground station H with parameters indicating a CTRL_CMD_HO (re = 0), including the protocol options, broadcast connection, ground station address filter and ATN router NETs parameters. |
| | 7 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 8 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK to new station H in response to the completed message. |
| | 9 | send | RF | INFO_a (s:= add_H; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst from the new ground station H to the station under test in order to test the existence of the new link. The INFO contains a unicast reservation for a response. |
| | | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 10 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_G) in the slot reserved by the INFO | IAa | Verify INFO_ACK received from the station under test addressed to station H, and therefore that the new link is activated. |
| | 11 | send | RF | INFO_a (s:= add_G; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst from the old ground station G to the station under test with M = 0. The INFO contains a unicast reservation for a response. |
| | 12 | verify | DLS | DM_DISC_a (s:= add_A; d:= add_G) in the slot reserved by the INFO | DMDCa | Verify that a DM_DISC is received from the station under test addressed to the old ground station G. |
| postamble | 13 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: LME_HO_GND_Broadcast_Invalid_A | | | | | | |
|---|------|--------|-----|---|-----|--|
| Purpose: To demonstrate that a mobile requested to comply with a ground requested broadcast handoff, where the mobile does not support broadcast handoff, will initiate either an air-initiated link handoff or request a link handoff. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | do | | CONFIGURE STATION UNDER TEST TO NOT SUPPORT GROUND REQUESTED BROADCAST HANDOFF | | |
| test body | 3 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 4 | send | RF | CTRL_RTS_a (s:= add_H; d:= add_A; IB:= 1; T:= 0; lg:= 2) | CRa | Send an RTS from ground station H. |
| | 5 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 6 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to ground station H. |
| | 7 | send | RF | UCTRL_b (s:= add_H; ucid:= 3; re:= 0; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; l:= 0; b:= 1; g:= add_G; lg:= 4; a:= add_A; lg:= 6; a:= ADM; r:= ARS) in the slot reserved by the CTS | UCb | Send a CTRL DLPDU from ground station H with parameters indicating a CTRL_CMD_HO (re = 0), including the protocol options, broadcast connection, ground station address filter and ATN router NETs parameters. |
| | 8 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 9 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK to station H in response to the completed message. |
| | 10 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 1) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to ground station H. The RTS contains a unicast reservation for a response. |
| | 11 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test from station H. |
| | 12 | verify | RF | CTRL_q (s:= add_A; d:= add_H; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS OR CTRL_d (s:= add_A; d:= add_H; M:= 0; T:= 1; re:= 0; c/r:= 0; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) | CTq | Verify that the station under test sends a CTRL DLPDU to station H with parameters indicating either a CTRL_CMD_HO (re = 1), including the protocol options parameter or sends a CTRL_CMD_HO (re = 0). |
| postamble | 13 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: LME_HO_GND_Broadcast_Invalid_B | | | | | | |
|--|------|--------|----------|---|-----|--|
| Purpose: To demonstrate that a mobile requested to comply with a ground requested broadcast handoff, where the ground station is not acceptable to the mobile, will initiate either an air-initiated link handoff or request a link handoff with a new ground station. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | Position | Input position of station under test as laT:= 0; lon:= E 22 NM | | Inform the station under test of its own position. |
| | 4 | send | App_In | Input ground track of station under test as 90 degrees (due East) | | Inform the station under test of its own ground track. |
| | 5 | send | RF | SYNC_BURST_a (s:=add_G; laT:= CPR_LAT(0); lon:= CPR_LON(E 5 NM)) | Sa | Send a sync burst from a current ground station G with position such that the station under test is moving away from it. |
| | 6 | send | RF | SYNC_BURST_a (s:=add_H; laT:= CPR_LAT(0); lon:= CPR_LON(W 170 NM)) | Sa | Send a sync burst from a simulated ground station H with position that is very far from the station under test and such that the station under test is moving away from it. |
| | 7 | send | RF | SYNC_BURST_a (s:=add_J; laT:= CPR_LAT(0); lon:= CPR_LON(E 30 NM)) | Sa | Send a sync burst from a simulated ground station J with position indicating that it is close to the station under test and in the direction in which the aircraft is heading. |
| | 8 | send | RF | CTRL_RTS_a (s:= add_H; d:= add_A; IB:= 1; T:= 0; lg:= 2) | CRa | Send an RTS from ground station H. |
| | 9 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 10 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to ground station H. |
| | 11 | send | RF | UCTRL_b (s:= add_H; ucid:= 3; re:= 0; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; l:= 0; b:= 1; g:= add_G; lg:= 4; a:= add_A; lg:= 6; a:= ADM; r:= ARS) in the slot reserved by the CTS | UCb | Send a CTRL DLPDU from ground station H with parameters indicating a CTRL_CMD_HO (re = 0), including the protocol options, broadcast connection, ground station address filter and ATN router NETs parameters. |
| | 12 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 13 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK to station H in response to the completed message. |
| | 14 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_J; IB:= 0; lg:= 1) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to ground station J. The RTS contains a unicast reservation for a response. |
| | 15 | send | RF | CTRL_CTS_a (s = add_J; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test from station J. |
| | 16 | verify | RF | CTRL_q (s:= add_A; d:= add_J; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS OR CTRL_d (s:= add_A; d:= add_J; M:= 0; T:= 1; re:= 0; c/r:= 0; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) | CTq | Verify that the station under test sends a CTRL DLPDU to station J with parameters indicating either a CTRL_CMD_HO (re = 1), including the protocol options parameter, or sends a CTRL_CMD_HO (re = 0). |
| postamble | 17 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: LME_HO_GND_Broadcast_Invalid_C | | | | | | |
|--|------|--------|-----|--|-------|--|
| Purpose: To demonstrate that a mobile requested to comply with a ground requested broadcast handoff, where the ground station address filter parameter does not equal the DLS address of a link that the mobile LME has, will not process the broadcast handoff. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | RF | CTRL_RTS_a (s:= add_H; d:= add_A; IB:= 1; T:= 0; lg:= 2) | CRa | Send an RTS from ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to ground station H. |
| | 6 | send | RF | UCTRL_b (s:= add_H; ucid:= 3; re:= 0; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; l:= 0; b:= 1; g:= add_J; lg:= 4; a:= add_A; lg:= 6; a:= ADM; r:= ARS) in the slot reserved by the CTS | UCb | Send a CTRL DLPDU from ground station H with parameters indicating a CTRL_CMD_HO (re = 0), including the protocol options, broadcast connection, ground station address filter and ATN router NETs parameters. The ground station address filter does not contain the address of ground station G, but that of another ground station J. |
| | 7 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 8 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK to station H in response to the completed message. |
| | 9 | send | RF | INFO_a (s:= add_H; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst from ground station H to the station under test in order to test the existence of the new link. The INFO contains a unicast reservation for a response. |
| | 10 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 11 | verify | DLS | DM_DISC_a (s:= add_A; d:= add_H) in the slot reserved by the INFO | DMDCa | Verify that a DM_DISC is received from the station under test addressed to the ground station H. |
| | 12 | send | RF | INFO_a (s:= add_G; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst from ground station G to the station under test with M = 0. The INFO contains a unicast reservation for a response. |
| | 13 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_G) in the slot reserved by the INFO | IAa | Verify that an INFO_ACK is received from the station under test addressed to station G, and therefore that the link with station G remains activated. |
| postamble | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: LME_HO_GND_Broadcast_Invalid_D | | | | | | |
|---|------|--------|-----|--|-------|---|
| Purpose: To demonstrate that a mobile requested to comply with a ground requested broadcast handoff, where no mobile identifier subfield in a broadcast connection parameter equals its mobile address, will not process the broadcast handoff. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | RF | CTRL_RTS_a (s:= add_H; d:= add_A; IB:= 1; T:= 0; lg:= 2) | CRa | Send an RTS from ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to ground station H. |
| | 6 | send | RF | UCTRL_b (s:= add_H; ucid:= 3; re:= 0; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; l:= 0; b:= 1; g:= add_G; lg:= 4; a:= add_C; lg:= 6; a:= ADM; r:= ARS) in the slot reserved by the CTS | UCb | Send a CTRL DLPDU from ground station H with parameters indicating a CTRL_CMD_HO (re = 0), including the protocol options, broadcast connection, ground station address filter and ATN router NETs parameters. The broadcast connection does not contain the address of the station under test, but that of another mobile C. |
| | 7 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 8 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK to station H in response to the completed message. |
| | 9 | send | RF | INFO_a (s:= add_H; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst from ground station H to the station under test in order to test the existence of the new link. The INFO contains a unicast reservation for a response. |
| | 10 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 11 | verify | DLS | DM_DISC_a (s:= add_A; d:= add_H) in the slot reserved by the INFO | DMDCa | Verify that a DM_DISC is received from the station under test addressed to the ground station H. |
| | 12 | send | RF | INFO_a (s:= add_G; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst from ground station G to the station under test with M = 0. The INFO contains a unicast reservation for a response. |
| | 13 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_G) in the slot reserved by the INFO | IAa | Verify that an INFO_ACK is received from the station under test addressed to station G, and therefore that the link with station G remains activated. |
| postamble | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: | | LME_TL1_Ground_Init | | | | |
|-----------------|------|--|-----|--|-------|--|
| Purpose: | | To demonstrate that a station displays correct operation of timer TL1 when handoff is initiated by the ground station. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | RF | CTRL_RTS_a (s:= add_H; d:= add_A; lB:= 1; T:= 0; lg:= 2) | CRa | Send an RTS from proposed new ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTS to new ground station H. |
| | 6 | send | RF | CTRL_h (s:= add_H; d:= add_A; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; l:= 1; lg:= 6; a:= ADM; r:= ARS; g:= add_H) in the slot reserved by the CTS | CTh | Send a CTRL DLPDU from new ground station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options, ATN router NETs, and ground-based system mask parameters. |
| | 7 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 8 | verify | RF | CTRL_ACK_CTRL_RTS_a (s = add_A; d = add_H; lg:= 2) in the ack slot reserved by the CTS | CACRa | Verify that a CTRL_ACK / CTRL_RTS is sent by the station under test to new ground station H. |
| | 9 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test. |
| | 10 | verify | RF | CTRL_d (s:= add_A; d:= add_H; M:= 0; T:= 0; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Verify that a CTRL burst indicating a CTRL_RSP_HO is sent by the station under test to new ground station H. |
| | 11 | record | RF | timestarT:= time at beginning of slot in which CTRL_d transmitted | CTd | Record the time at which the CTRL_RSP_HO was transmitted (this is the time at which timer TL1 is started). |
| | 12 | send | RF | CTRL_ACK_a (s:= add_H; d:= add_A) in the ack slot reserved by the CTS | CAa | Send the ACK from new station H in response to the completed message. |
| | 13 | rep 5 | | m:= {20, 30, 40, 50, 59}; info:= {info1, info2, info3, info4, info5} | | Start loop to send packets on the old link before TL1 expires. All the "infox" packets require a short transmission. |
| | 14 | await | | time:= timestart + m | | Wait for a specified time after the CTRL_RSP_HO was transmitted. |
| | 15 | send | RF | INFO_a (s:= add_G; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst from the old ground station G to the station under test with M = 0. The INFO contains a unicast reservation for a response. |
| | 16 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_G) in the slot reserved by the INFO | IAa | Verify INFO_ACK received from the station under test addressed to old station G. |
| | 17 | endrep | | | | End loop. |
| | 18 | rep 4 | | m:= {60, 70, 80, 90}; info:= {info6, info7, info8, info9} | | Start loop to send packets on the old link after TL1 has expired. |
| | 19 | await | | time:= timestart + m | | Wait for a specified time after the CTRL_RSP_HO was transmitted. |

| | | | | | | |
|------------------------|-------------|---|------------|---|------------|--|
| Test case name: | | LME_TL1_Ground_Init | | | | |
| Purpose: | | To demonstrate that a station displays correct operation of timer TL1 when handoff is initiated by the ground station. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 20 | send | RF | INFO_a (s:= add_G; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst from the old ground station G to the station under test with M = 0. The INFO contains a unicast reservation for a response. |
| | 21 | verify | DLS | DM_DISC_a (s:= add_A; d:= add_G) in the slot reserved by the INFO | DMDCa | Verify DM_DISC received from station A addressed to old station G. |
| | 22 | endrep | | | | End loop. |
| postamble | 23 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| | | | | | | |
|------------------------|-------------|---|------------|---|------------|--|
| Test case name: | | LME_TL1_Mob_Init | | | | |
| Purpose: | | To demonstrate that a station displays correct operation of timer TL1 when handoff is initiated by the mobile. | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | send | LME | REQUEST TO TRANSMIT MOBILE-INITIATED HANDOFF to new ground station H | | Send a request at the LME of the station under test to transmit a mobile-initiated handoff to new ground station H. |
| | 4 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 5 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to new ground station H. The RTS contains a unicast reservation for a response. |
| | 6 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS to the station under test from new ground station H. |
| | 7 | verify | RF | CTRL_q (s:= add_A; d:= add_H; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS | CTq | Verify that the station under test sends a CTRL DLPDU to new ground station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options parameter. |
| | 8 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_H; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS from new ground station H. |
| | 9 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 10 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to new ground station H. |
| | 11 | send | RF | CTRL_d (s:= add_H; d:= add_A; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Send a CTRL burst indicating a CTRL_RSP_HO from new ground station H. |
| | 12 | record | LME | timestarT:= time at beginning of slot in which CTRL_d received by station under test | CTd | Record the time at which the CTRL_RSP_HO was received by the station under test (this is the time at which timer TL1 is started). |
| | 13 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK to new ground station H in response to the completed message. |
| | 14 | rep 5 | | m:= {2, 3, 14, 18, 19}; info:= {info1, info2, info3, info4, info5} | | Start loop to send packets on the old link before TL1 expires. All the "infox" packets require a short transmission. |
| | 15 | await | | time:= timestart + m | | Wait for a specified time after the CTRL_RSP_HO was transmitted. |

| Test case name: LME_TL1_Mob_Init | | | | | | |
|---|------|--------|-----|--|-------|--|
| Purpose: To demonstrate that a station displays correct operation of timer TL1 when handoff is initiated by the mobile. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 16 | send | RF | INFO_a (s:= add_G; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst to the station under test from old ground station G with M = 0. The INFO contains a unicast reservation for a response. |
| | 17 | verify | RF | INFO_ACK_a (s:= add_A; d:= add_G) in the slot reserved by the INFO | IAa | Verify that an INFO_ACK addressed to the old ground station G is received from the station under test. |
| | 18 | endrep | | | | End loop. |
| | 19 | rep 4 | | m:= {20, 30, 40, 50}; info:= {info6, info7, info8, info9} | | Start loop to send packets on the old link after TL1 has expired. |
| | 20 | await | | time:= timestart + m | | Wait for a specified time after the CTRL_RSP_HO was transmitted. |
| | 21 | send | RF | INFO_a (s:= add_G; d:= add_A; in:= info; M:= 0; pr:= 3) | IFa | Send an INFO burst to the station under test from old ground station G with M = 0. The INFO contains a unicast reservation for a response. |
| | 22 | verify | DLS | DM_DISC_a (s:= add_A; d:= add_G) in the slot reserved by the INFO | DMDCa | Verify DM_DISC addressed to the old ground station G is received from the station under test. |
| | 23 | endrep | | | | End loop. |
| postamble | 24 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test case name: LME_L1_Mob | | | | | | |
|--|------|--------|-------|--|-----|---|
| Purpose: To demonstrate that a station displays correct operation of counter L1. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_LME_INITIALIZE_LINK_MOB | | Perform link establishment initiated by the mobile (with ground station G). |
| | 3 | rep 3 | | infoshort:= {info1, info2, info3} | | Start loop to send a few data packets to establish that data is being sent normally. Both the "infox" packets require a short transmission. |
| | 4 | send | Appln | REQUEST TO TRANSMIT DATA to ground station G (in:= infoshort) | | Send a request to the station under test to transmit data to ground station G. |
| | 5 | verify | RF | INFO_a (s:= add_A; d:= add_G; in:= infoshort; M:= 0) | IFa | Verify that the station under test sends an INFO burst to station G. The INFO contains a unicast reservation for a response. |
| | | send | RF | INFO_ACK_a (s:= add_G; d:= add_A) in the slot reserved by the INFO | IAa | Send the ACK from station G in response. |
| | 6 | endrep | | | | End loop. |
| | 7 | send | Appln | REQUEST TO TRANSMIT DATA to station G (in:= infolong) | | Send a request to the station under test to transmit data to station G. The "info" packet requires a long transmission, and therefore an RTS will be generated. |
| | 8 | verify | RF | INFO_RTS_a (s:= add_A; d:= add_G) transmitted | | Verify that an RTS is transmitted. The RTS contains a unicast reservation for a response. |
| | 9 | rep 3 | | n:= 1 | | Start loop. |
| | 10 | record | RF | timestart:= time at beginning of slot in which last INFO_RTS_a transmitted | IRa | Record the time at which the last INFO_RTS_a was transmitted. |

| Test case name: LME_L1_Mob | | | | | | |
|--|------|--------|-----|---|-------|--|
| Purpose: To demonstrate that a station displays correct operation of counter L1. | | | | | | |
| Context | Step | Action | PCO | Action qualifier | Ref | Comment |
| | 11 | await | | time:= timestart + 5 | | Wait for TL3 seconds specified time after the last RTS was transmitted. |
| | 12 | verify | DLS | INFO_RTS_a (s:= add_A; d:= add_B) transmitted | | Verify at the DLS that an RTS is re-transmitted. The RTS contains a unicast reservation for a response. |
| | 13 | endrep | | | | End loop. |
| | 15 | verify | RF | CTRL_RTS_a (s:= add_A; d:= add_H; IB:= 0; lg:= 2) | CRa | Verify that the station under test sends an RTS using the long transmission procedures to simulated new ground station H. The RTS contains a unicast reservation for a response. |
| | 16 | send | RF | CTRL_CTS_a (s = add_H; d = add_A) in the slot reserved by the RTS | CCa | Send a CTRL_CTS from station H to the station under test. |
| | 17 | verify | RF | CTRL_q (s:= add_A; d:= add_H; re:= 1; c/r:= 0; lg:= 1; h:= 1; r:= 0; rrr:= 0; sss:= 0; b:=0; i:= 1) in the slot reserved by the CTS | CTq | Verify that the station under test sends a CTRL DLPDU to station H with parameters indicating a CTRL_CMD_HO (re = 1), including the protocol options parameter. |
| | 18 | send | RF | CTRL_ACK_CTRL_RTS_a (s = add_H; d = add_A; lg:= 2) in the ack slot reserved by the CTS | CACRa | Send a CTRL_ACK / CTRL_RTS from station H. |
| | 19 | await | RF | BURST from station under test | | Wait for a burst from the station under test. |
| | 20 | verify | RF | CTRL_CTS_a (s = add_A; d = add_H) in the slot reserved by the RTS | CCa | Verify that the station under test sends a CTRL_CTS to station H. |
| | 21 | send | RF | CTRL_d (s:= add_H; d:= add_A; re:= 1; c/r:= 1; h:= 1; r:= 0; lg:= 1; rrr:= 0; sss:= 0) in the slot reserved by the CTS | CTd | Send a CTRL burst indicating a CTRL_RSP_HO from station H. |
| | 22 | verify | RF | CTRL_ACK_a (s:= add_A; d:= add_H) in the ack slot reserved by the CTS | CAa | Verify that the station under test sends an ACK in response to the completed message. |
| postamble | 23 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

Annex A (informative):

Cross reference matrix

Table A.1 outlines the mapping between the VDL Mode 4 airborne station requirements and the related test procedures. The table also provides a cross reference to the ICAO reference material from which many of the requirements within the present document are derived. In these tables:

- column 1 is a reference to the requirement in the present document;
- column 2 is a reference to the equivalent requirements in ICAO VDL4 Technical Manual [1];
- column 3 identifies individual requirements within ICAO VDL4 Technical Manual [1];
- column 4 identifies clause titles taken from the present document;
- column 5 is a reference to testing requirements specified elsewhere in the present document. Several tests verify a whole group of requirements. They are only mentioned in the first row of such a group, usually a headline. The applicability of these tests to the subordinated requirements is indicated by ditto marks ("") in the rows following the first instance of a test case name. Amplification of individual entries is provided by the following notes;
- column 6 qualifies each test procedure to be:
 - Essential: meaning that it is included with the Essential Radio Test Suite and therefore the requirement should be demonstrated to be met in accordance with the referenced procedures.
 - Other: meaning that the test procedure is illustrative but other means of demonstrating compliance with the requirement are permitted.
 - eXcluded: meaning that there is no specific test for the requirement.

NOTE 1: The clause number in column 1 is a headline or an introduction to requirements that are detailed in subsequent clauses. No test can be applied.

NOTE 1a: The clause number in column 1 is a definition. No test can be applied.

NOTE 2: The requirement listed in column 1 does not allow definition of a satisfactory go/no go test, for example, because it would be technically infeasible, or economically unreasonable. There are circumstances where the implementer can provide reasoned argument or test evidence that the implementation under test does conform to the requirements in column 1. For each of these circumstances the implementer may be required to satisfy the authorities by separate technical evidence.

NOTE 3: The requirement listed in column 1 is applicable only to VDL Mode 4 ground equipment. No mobile equipment test is required.

NOTE 4: This topic is heavily dependent on the implementation or results from a recommendation. No particular test is therefore provided in the present document.

NOTE 5: All tests whether "E" or "O" are relevant to the requirements. Rows designated "E" collectively make up the Essential Radio Test Suite; those designated "O" make up the Other Test Suite; for those designated "X" there is no test specified corresponding to the requirement. All tests classified "E" should be performed as specified with satisfactory outcomes as a necessary condition for a presumption of conformity. Requirements associated with tests classified "O" or "X" should be complied with as a necessary condition for presumption of conformity, although conformance with the requirement may be claimed by an equivalent test or by manufacturer's assertion supported by appropriate entries in the technical construction file.

Table A.1: VDL Mode 4 requirements according to ICAO VDL4 Technical Manual [1]

| Requirement reference | Reference in ICAO Doc 9816 [1] | Req | Title | Test case | E/O/X |
|-----------------------|--------------------------------|-----|---|---|-------|
| 5.1 | 1.4 | | DLS SUBLAYER | see note 1 | X |
| 5.1.1 | | | General | see note 1 | X |
| 5.1.1.1 | 1.4.1 | | Services | see note 1 | X |
| 5.1.1.1.1 | 1.1.2.1 | a | | see note 2 | O |
| 5.1.1.1.2 | 1.1.2.2 | a | | see note 2 | O |
| 5.1.1.1.3 | 1.4 | a | | see note 2 | O |
| 5.1.1.1.4 | 1.4.1.1.1 | a | | see note 2 | O |
| 5.1.1.1.5 | 1.4.1.1.3 | a | | see note 2 | O |
| 5.1.1.1.6 | 1.4.1.1.4 | a | | DLS_Links | E |
| 5.1.1.2 | 1.4.1.2 | | Data transfer | see note 1 | X |
| 5.1.1.2.1 | 1.4.1.2 | a | | see note 2 | O |
| 5.1.1.2.2 | 1.4.1.2 | b | | see note 1a | X |
| 5.1.1.2.3 | 1.4.1.2 | c | | DLS_ND3 | E |
| 5.1.1.2.4 | 1.4.1.2 | d | | DLS_ND3 | E |
| 5.1.1.2.5 | 1.4.1.2 | e | | DLS_Data | E |
| 5.1.1.3 | 1.4.1.3 | | DATA DLPDU duplicate suppression and sequencing | see note 1 | X |
| 5.1.1.3.1 | 1.4.1.3 | a | | DLS_Duplicate_Short DLS_Long_Order | E |
| 5.1.1.4 | 1.4.1.4 | | Error detection | see note 1 | X |
| 5.1.1.4.1 | 1.4.1.4 | a | | see note 2 | O |
| 5.1.1.5 | 1.4.1.5 | | Station identification | see note 1 | X |
| 5.1.1.5.1 | 1.4.1.5 | a | | see note 2 | O |
| 5.1.1.6 | 1.4.1.6 | | Broadcast addressing | see note 1 | X |
| 5.1.1.6.1 | 1.4.1.6 | a | | see note 2 | O |
| 5.1.1.7 | 1.4.1.7 | | DLS Priority | see note 1 | X |
| 5.1.1.7.1 | 1.4.1.7 | a | | see note 2 | O |
| 5.1.1.8 | 1.4.1.8 | | DLS link control DLPDUs | see note 1 | X |
| 5.1.1.8.1 | 1.4.1.8 | a | | see note 2 | O |
| 5.1.2 | 1.4.2 | | DLS protocol specification | see note 1 | X |
| 5.1.2.1 | 1.4.2.1 | | State Variables | see note 1 | X |
| 5.1.2.1.1 | 1.4.2.1 | a | | see note 1a | X |
| 5.1.2.2 | 1.4.2.3 | | DLS burst formats | see note 1 | X |
| 5.1.2.2.1 | 1.4.2.3.1 | a | | see note 1a | X |
| 5.1.2.2.2 | 1.4.2.3.1.2 | a | | see note 1a | X |
| 5.1.2.2.3 | 1.4.2.3.1.2 | b | | see note 1a | X |
| 5.1.2.2.4 | 1.4.2.3.2.1 | a | | see note 1a | X |
| 5.1.2.2.5 | 1.4.2.3.2.2 | a | | see note 1a | X |
| 5.1.2.2.6 | 1.4.2.3.2.2 | b | | see note 1a | X |
| 5.1.2.2.7 | 1.4.2.3.2.4 | a | | see note 1a | X |
| 5.1.2.2.8 | 1.4.2.3.2.4 | b | | see note 1a | X |
| 5.1.2.2.9 | 1.4.2.3.2.5 | a | | see note 1a | X |
| 5.1.2.2.10 | 1.4.2.3.2.5 | b | | see note 1a | X |
| 5.1.2.2.11 | 1.4.2.3.2.5.1 | c | | see note 1a | X |
| 5.1.2.2.12 | 1.4.2.3.2.5.1 | a | | see note 2 | O |
| 5.1.2.2.13 | 1.4.2.3.2.5.1 | b | | see note 2 | O |
| 5.1.2.2.14 | 1.4.2.3.3 | a | | DLS_Long_TM_Send_A DLS_Long_TM_Send_B DLS_Long_T_Receive_A DLS_Long_T_Receive_B DLS_Short_T_Receive DLS_Short_T_Send | E |
| 5.1.2.2.15 | 1.4.2.3.3 | b | | DLS_Initialize_NSCOP_Send | E |
| 5.1.2.2.16 | 1.4.2.3.4 | a | | DLS_Long_TM_Send_A DLS_Long_TM_Send_B | E |
| 5.1.2.2.17 | 1.4.2.3.5 | a | | see note 1a | X |
| 5.1.2.2.18 | 1.4.2.3.6 | a | | DLS_Length | E |
| 5.1.2.2.19 | 1.4.2.3.6 | b | | see note 1a | X |
| 5.1.2.2.20 | 1.4.2.3.7 | a | | see note 2 | O |
| 5.1.2.2.20a | 1.4.2.3.8 | a | | see note 1a | X |
| 5.1.2.2.20b | 1.4.2.3.9 | a | | see note 1a | X |

| Requirement reference | Reference in ICAO Doc 9816 [1] | Req | Title | Test case | E/O/X |
|-----------------------|--------------------------------|-----|--|--|-------|
| 5.1.2.2.21 | 1.4.2.3.11 | a | | see note 2 | O |
| 5.1.2.2.22 | 1.4.2.3.11 | b | | see note 2 | O |
| 5.1.2.2.23 | 1.4.2.3.12 | a | | see note 2 | O |
| 5.1.2.2.24 | 1.4.2.3.12 | b | | see note 2 | O |
| 5.1.3 | 1.4.3 | | DLS system parameters | see note 1 | X |
| 5.1.3.1 | 1.4.3 | a | | see note 1a | X |
| 5.1.3.2 | 1.4.3 | b | | see note 2 | O |
| 5.1.3.2a | 1.4.3 | c | | see note 2 | O |
| 5.1.3.2b | 1.4.3.1 | a | | DLS_TD1_Link | E |
| 5.1.3.2c | 1.4.3.1 | b | | DLS_TD1_Link | E |
| 5.1.3.2d | 1.4.3.1 | c | | DLS_TD1_Link | E |
| 5.1.3.2e | 1.4.3.2 | a | | DLS_TD1_TD2_Link | E |
| 5.1.3.2f | 1.4.3.2 | b | | DLS_TD1_TD2_Link | E |
| 5.1.3.2g | 1.4.3.2 | c | | DLS_TD1_TD2_Link | E |
| 5.1.3.3 | 1.4.3.3 | a | | see note 1a | X |
| 5.1.3.4 | 1.4.3.3 | b | | DLS_ND1_Short_Receive | E |
| 5.1.3.5 | 1.4.3.3 | c | | DLS_ND1_Short_Send | E |
| 5.1.3.6 | 1.4.3.4.1 | a | | DLS_ND2 | E |
| 5.1.3.7 | 1.4.3.4.2 | a | | see note 1a | X |
| 5.1.3.8 | 1.4.3.5 | a | | DLS_ND3 | E |
| 5.1.4 | 1.4.4 | | DLS procedures | see note 1 | X |
| 5.1.4.1 | 1.4.4.2 | | Setting of re-transmission parameter | see note 1 | X |
| 5.1.4.1.1 | 1.4.4.2 | a | | DLS_Short_NoAck | E |
| 5.1.4.2 | 1.4.4.3 | | Selection of user data packet for transmission | see note 1 | X |
| 5.1.4.2.1 | 1.4.4.3.1.1 | a | | DLS_Priority_Long DLS_Priority_Long_Short DLS_Priority_Short | E |
| 5.1.4.2.2 | 1.4.4.3.1.2 | a | | DLS_Priority_Long DLS_Priority_Long_Short DLS_Priority_Short | E |
| 5.1.4.2.3 | 1.4.4.3.1.3 | a | | DLS_Priority_CTRL | E |
| 5.1.4.2.4 | 1.4.4.3.1.4 | a | | DLS_Priority_INFO_RTS DLS_Long_UDATA_Send | E |
| 5.1.4.2.5 | 1.4.4.3.2.1.1 | a | | DLS_ND3 | E |
| 5.1.4.2.6 | 1.4.4.3.2.1.1 | b | | DLS_ND3 | E |
| 5.1.4.2.7 | 1.4.4.3.2.1.2 | c | | see note 2 | O |
| 5.1.4.2.8 | 1.4.4.3.2.2 | a | | DLS_ND3 DLS_Short_T_Send | E |
| 5.1.4.2.9 | 1.4.4.3.2.2 | b | | DLS_Long_TM_Send_A DLS_Priority_Long DLS_Priority_Long_Short DLS_Priority_Short DLS_Short_T_Send | E |
| 5.1.4.2.10 | 1.4.4.3.2.2 | c | | DLS_Priority_Long DLS_Priority_Long_Short DLS_Priority_Short DLS_Short_T_Send | E |
| 5.1.4.2.11 | 1.4.4.3.2.3 | a | | DLS_ND3 DLS_Long_TM_Send_B | E |
| 5.1.4.2.12 | 1.4.4.3.2.3 | b | | DLS_Long_TM_Send_B | E |
| 5.1.4.2.13 | 1.4.4.3.2.3 | c | | DLS_Long_TM_Send_B | E |
| 5.1.4.2.14 | 1.4.4.3.2.3 | d | | DLS_Long_TM_Send_B | E |
| 5.1.4.2.15 | 1.4.4.3.3.1 | a | | DLS_Initialize_Send_Long | E |
| 5.1.4.2.16 | 1.4.4.3.3.1 | b | | DLS_Initialize_Send_Long | E |
| 5.1.4.2.17 | 1.4.4.3.3.1.1 | a | | DLS_Initialize_Receive_Long | E |
| 5.1.4.2.18 | 1.4.4.3.3.1.1 | b | | DLS_Initialize_Send_Long DLS_Initialize_Receive_Long | E |
| 5.1.4.2.19 | 1.4.4.3.3.1.1 | c | | see note 2 | O |
| 5.1.4.2.20 | 1.4.4.3.3.1.1 | d | | see note 2 | O |
| 5.1.4.2.21 | 1.4.4.3.3.1.1 | e | | see note 2 | O |
| 5.1.4.2.22 | 1.4.4.3.3.1.1 | f | | see note 2 | O |
| 5.1.4.2.23 | 1.4.4.3.3.1.2 | a | | see note 2 | O |
| 5.1.4.2.24 | 1.4.4.3.3.1.2 | b | | see note 2 | O |

| Requirement reference | Reference in ICAO Doc 9816 [1] | Req | Title | Test case | E/O/X |
|-----------------------|--------------------------------|-----|--------------------------------------|--|-------|
| 5.1.4.2.25 | 1.4.4.3.3.1.2 | c | | see note 2 | O |
| 5.1.4.2.26 | 1.4.4.3.3.1.3 | a | | DLS_Long_IB_Error | E |
| 5.1.4.2.27 | 1.4.4.3.3.1.3 | b | | DLS_Long_NoLink DLS_Short_NoLink | E |
| 5.1.4.2.28 | 1.4.4.3.3.1.4 | a | | DLS_Long_SZOM_Error DLS_Short_SZOM_Error | E |
| 5.1.4.2.28a | 1.4.4.3.3.2.1 | a | | DLS_INFO_SZOM_Send DLS_INFO_RTS_SZOM_Send | E |
| 5.1.4.2.28b | 1.4.4.3.3.2.1 | b | | DLS_INFO_SZOM_Send DLS_INFO_RTS_SZOM_Send | E |
| 5.1.4.2.28c | 1.4.4.3.3.2.2 | a | | DLS_INFO_RTS_SZOM_Receive DLS_INFO_SZOM_Receive | E |
| 5.1.4.2.28d | 1.4.4.3.3.2.2 | a | | DLS_INFO_RTS_SZOM_Receive DLS_INFO_SZOM_Receive | E |
| 5.1.4.2.28e | 1.4.4.3.3.2.2 | b | | DLS_INFO_RTS_SZOM_Receive DLS_INFO_SZOM_Receive | E |
| 5.1.4.2.28f | 1.4.4.3.3.2.2 | c | | see note 2 | O |
| 5.1.4.2.28g | 1.4.4.3.3.2.3 | a | | see note 2 | O |
| 5.1.4.2.28h | 1.4.4.3.3.2.4 | a | | DLS_SZOM_Retrans | E |
| 5.1.4.2.28i | 1.4.4.3.3.2.4 | b | | DLS_SZOM_Retrans | E |
| 5.1.4.2.29 | 1.4.4.3.3.3 | a | | see note 2 | O |
| 5.1.4.3 | 1.4.4.4 | | Selection of transmission procedures | see note 1 | X |
| 5.1.4.3.1 | 1.4.4.4 | a | | DLS_ND2 | E |
| 5.1.4.3.2 | 1.4.4.4 | b | | see note 1a | X |
| 5.1.4.3.3 | 1.4.4.4 | c | | see note 2 | O |
| 5.1.4.3.4 | 1.4.4.4 | d | | DLS_ND2 | E |
| 5.1.4.3.5 | 1.4.4.4 | e | | DLS_ND2 | E |
| 5.1.4.3.6 | 1.4.4.4 | f | | DLS_ND2 DLS_Short_T_Send | E |
| 5.1.4.3.7 | 1.4.4.4.1 | a | | see note 4 | O |
| 5.1.4.4 | 1.4.4.5 | | Short transmission procedures | see note 1 | X |
| 5.1.4.4.1 | 1.4.4.5.1.1 | a | | DLS_Short_T_Send | E |
| 5.1.4.4.2 | 1.4.4.5.1.1 | b | | DLS_Short_T_Send | E |
| 5.1.4.4.3 | 1.4.4.5.1.2 | a | | DLS_Short_T_Send | E |
| 5.1.4.4.4 | 1.4.4.5.2.1 | a | | DLS_Short_T_Receive | E |
| 5.1.4.4.5 | 1.4.4.5.2.1 | b | | DLS_Short_T_Receive | E |
| 5.1.4.4.6 | 1.4.4.5.2.1 | c | | DLS_Short_T_Receive | E |
| 5.1.4.4.7 | 1.4.4.5.3 | a | | DLS_Short_NoAck | E |
| 5.1.4.5 | 1.4.4.6 | | Long transmission procedures | see note 1 | X |
| 5.1.4.5.1 | 1.4.4.6.1 | a | | DLS_ND2 DLS_Long_TM_Send_A DLS_Long_TM_Send_B DLS_Initialize_Send_Long DLS_Long_UDATA_Send | E |
| 5.1.4.5.2 | 1.4.4.6.1 | b | | DLS_Long_TM_Send_A DLS_Long_TM_Send_B DLS_Initialize_Send_Long DLS_Long_UDATA_Send | E |
| 5.1.4.5.3 | 1.4.4.6.1 | c | | DLS_Priority_INFO_RTS DLS_Long_UDATA_Send | E |
| 5.1.4.5.4 | 1.4.4.6.1 | d | | DLS_Length DLS_Initialize_Send_Long DLS_Long_UDATA_Send | E |
| 5.1.4.5.5 | 1.4.4.6.1 | e | | DLS_Long_TM_Send_A DLS_Long_TM_Send_B DLS_Initialize_Send_Long | E |
| 5.1.4.5.6 | 1.4.4.6.1 | f | | DLS_Long_TM_Send_A DLS_Long_TM_Send_B DLS_Initialize_Send_Long DLS_Long_UDATA_Send | E |
| 5.1.4.5.7 | 1.4.4.6.3.1.1 | a | | DLS_Long_T_Receive_A DLS_Initialize_Receive_Long DLS_Long_UDATA_Receive | E |

| Requirement reference | Reference in ICAO Doc 9816 [1] | Req | Title | Test case | E/O/X |
|-----------------------|--------------------------------|-----|---------------------------------|---|-------|
| 5.1.4.5.8 | 1.4.4.6.3.1.2 | a | | DLS_Long_T_Receive_A DLS_Initialize_Receive_Long DLS_Long_UDATA_Receive | E |
| 5.1.4.5.9 | 1.4.4.6.3.1.3 | a | | DLS_Long_T_Receive_A DLS_Initialize_Receive_Long | E |
| 5.1.4.5.10 | 1.4.4.6.3.1.4 | a | | DLS_Long_UDATA_Receive | E |
| 5.1.4.5.11 | 1.4.4.6.3.2 | a | | DLS_Long_T_NoAck_A | E |
| 5.1.4.5.12 | 1.4.4.6.3.2 | b | | DLS_Long_T_NoAck_A | E |
| 5.1.4.5.13 | 1.4.4.6.3.2 | c | | DLS_Long_T_NoAck_A | E |
| 5.1.4.5.14 | 1.4.4.6.3.2 | d | | DLS_Long_T_NoAck_A | E |
| 5.1.4.5.15 | 1.4.4.6.3.3 | a | | DLS_Long_Busy_A | E |
| 5.1.4.5.16 | 1.4.4.6.4.1.1 | a | | DLS_Long_TM_Send_A DLS_Long_TM_Send_B DLS_Initialize_Send_Long | E |
| 5.1.4.5.17 | 1.4.4.6.4.1.1 | b | | DLS_Long_TM_Send_A DLS_Long_TM_Send_B DLS_Initialize_Send_Long | E |
| 5.1.4.5.18 | 1.4.4.6.4.1.2 | a | | DLS_Long_UDATA_Send | E |
| 5.1.4.5.19 | 1.4.4.6.4.2 | a | | DLS_Long_NoInfo_A | E |
| 5.1.4.5.20 | 1.4.4.6.4.3 | a | | see note 4 | O |
| 5.1.4.5.21 | 1.4.4.6.5.1 | a | | DLS_Long_T_Receive_A DLS_Initialize_Receive_Long | E |
| 5.1.4.5.22 | 1.4.4.6.5.2.1 | a | | DLS_Long_NACK_A | E |
| 5.1.4.5.23 | 1.4.4.6.5.2.1 | b | | DLS_Long_NACK_A | E |
| 5.1.4.5.24 | 1.4.4.6.5.2.1 | c | | DLS_Long_NACK_A | E |
| 5.1.4.5.25 | 1.4.4.6.5.2.2 | a | | DLS_Long_NACK_A | E |
| 5.1.4.6 | 1.4.4.8 | | No link with sender | see note 1 | X |
| 5.1.4.6.1 | 1.4.4.8 | a | | DLS_Long_NoLink DLS_Short_NoLink | E |
| 5.1.4.6.2 | 1.4.4.8 | b | | DLS_Long_NoLink DLS_Short_NoLink | E |
| 5.1.4.7 | 1.4.4.9 | | User data packet reception | see note 1 | X |
| 5.1.4.7.1 | 1.4.4.9.1.1 | a | | DLS_Short_T_Receive | E |
| 5.1.4.7.2 | 1.4.4.9.1.1 | b | | DLS_Short_T_Receive DLS_INFO_SZOM_Receive | E |
| 5.1.4.7.3 | 1.4.4.9.1.1 | c | | DLS_Duplicate_Short DLS_Duplicate_Long | E |
| 5.1.4.7.4 | 1.4.4.9.2.1 | a | | DLS_Long_Order | E |
| 5.1.4.7.5 | 1.4.4.9.2.2 | a | | DLS_Long_Order | E |
| 5.1.4.7.6 | 1.4.4.9.3 | a | | see note 2 | O |
| 5.1.4.8 | 1.4.4.10 | | Receipt of ACK DLPDU | see note 1 | X |
| 5.1.4.8.1 | 1.4.4.10.1.1 | a | | DLS_Short_T_Receive DLS_Long_T_Receive_A | E |
| 5.1.4.8.2 | 1.4.4.10.1.2 | a | | DLS_Short_T_Receive DLS_Long_T_Receive_A | E |
| 5.1.4.8.3 | 1.4.4.10.1.3 | a | | DLS_Short_NoAck | E |
| 5.1.4.8.4 | 1.4.4.10.1.4 | a | | DLS_Priority_Long_Short | E |
| 5.1.4.8.5 | 1.4.4.10.2 | a | | see note 2 | O |
| 5.1.4.8.6 | 1.4.4.10.2 | b | | see note 2 | O |
| 5.1.4.8.7 | 1.4.4.10.2 | c | | see note 2 | O |
| 5.1.4.9 | 1.4.4.11 | | Link reset | see note 1 | X |
| 5.1.4.9.1 | 1.4.4.11.1 | a | | see note 2 | O |
| 5.1.4.9.2 | 1.4.4.11.1 | b | | see note 2 | O |
| 5.1.4.9.3 | 1.4.4.11.2.1 | a | | see note 2 | O |
| 5.1.4.9.4 | 1.4.4.11.2.1 | b | | see note 2 | O |
| 5.1.4.9.5 | 1.4.4.11.2.1 | c | | see note 2 | O |
| 5.1.4.9.6 | 1.4.4.11.2.1 | d | | see note 2 | O |
| 5.1.4.9.7 | 1.4.4.11.2.1 | e | | see note 2 | O |
| 5.1.4.9.8 | 1.4.4.11.2.1 | f | | see note 2 | O |
| 5.1.4.9.9 | 1.4.4.11.2.2 | a | | see note 2 | O |
| 5.1.4.10 | 1.4.4.12 | | Linking DLS DLPDU transmissions | see note 1 | X |
| 5.1.4.10.1 | 1.4.4.12 | a | | see note 2 | O |
| 5.1.4.10.2 | 1.4.4.12.1 | a | | see note 4 | O |

| Requirement reference | Reference in ICAO Doc 9816 [1] | Req | Title | Test case | E/O/X |
|-----------------------|--------------------------------|-----|--|---|-------|
| 5.1.4.10.3 | 1.4.4.12.2 | a | | DLS_Long_TM_Send_B DLS_Long_T_Receive_B DLS_Short_DATA_ACK DLS_Long_DATA_ACK | E |
| 5.1.4.10.4 | 1.4.4.12.3 | a | | see note 4 | O |
| 5.1.4.10.5 | 1.4.4.12.4.1 | a | | DLS_Long_T_Receive_B | E |
| 5.1.4.10.6 | 1.4.4.12.4.1 | b | | DLS_Long_T_Receive_B | E |
| 5.1.4.10.7 | 1.4.4.12.4.1 | c | | DLS_Long_T_Receive_B | E |
| 5.1.4.10.8 | 1.4.4.12.4.2 | a | | DLS_Long_T_Receive_B | E |
| 5.1.4.10.9 | 1.4.4.12.5 | a | | DLS_Short_DATA_ACK | E |
| 5.1.4.10.10 | 1.4.4.12.5 | b | | DLS_Short_DATA_ACK | E |
| 5.1.4.10.11 | 1.4.4.12.6 | a | | DLS_Long_DATA_ACK | E |
| 5.1.4.10.12 | 1.4.4.12.7 | a | | see note 4 | O |
| 5.1.4.11 | 1.4.4.13 | | CTRL DLPDU | see note 1 | X |
| 5.1.4.11.1 | 1.4.4.13 | a | | see note 2 | O |
| 5.1.4.12.1 | 1.4.4.14 | a | | see note 1a | X |
| 5.1.4.12.2 | 1.4.4.14.1 | a | | DLS_Short_T_Send DLS_Short_T_Receive DLS_Long_TM_Send_A DLS_Long_TM_Send_B DLS_Long_T_Receive_A | E |
| 5.1.4.12.3 | 1.4.4.14.2.1 | a | | DLS_TD1_Link | E |
| 5.1.4.12.4 | 1.4.4.14.2.2 | a | | DLS_TD1_TD2_Link | E |
| 5.1.4.12.5 | 1.4.4.14.2.3 | a | | DLS_Link_Terminated DLS_SZOM_Retrans | E |
| 5.1.4.12.6 | 1.4.4.14.3.1 | a | | DLS_ZOCOP_Link_CTRL | E |
| 5.1.4.12.7 | 1.4.4.14.3.2 | a | | see note 2 | O |
| 5.2 | 1.5 | | LINK MANAGEMENT ENTITY SUBLAYER | see note 1 | X |
| 5.2.1 | 1.5.1 | | Services | see note 1 | X |
| 5.2.1.1 | | | General | see note 1 | X |
| 5.2.1.1.1 | 1.5.1 | a | | see note 1a | X |
| 5.2.1.2 | 1.5.1.1 | | Link provision | see note 1 | X |
| 5.2.1.2.1 | 1.5.1.1.1 | a | | see note 2 | O |
| 5.2.1.2.2 | 1.5.1.1.1 | b | | see note 2 | O |
| 5.2.1.2.2a | 1.5.1.1.1 | c | | see note 2 | O |
| 5.2.1.2.4 | 1.5.1.1.2 | a | | see note 2 | O |
| 5.2.1.2.5a | 1.5.1.1.2 | c | | see note 2 | O |
| 5.2.1.2.6 | 1.5.1.1.2 | d | | see note 2 | O |
| 5.2.1.2.7a | 1.5.1.1.2 | f | | see note 2 | O |
| 5.2.1.2.8 | 1.5.1.1.3 | a | | see note 2 | O |
| 5.2.1.3 | 1.5.1.2 | | Link change notifications | see note 1 | X |
| 5.2.1.3.1 | 1.5.1.2 | a | | see note 2 | O |
| 5.2.1.4 | 1.5.2.5 | | CTRL DLPDU | see note 1 | X |
| 5.2.1.4.1 | 1.5.2.5.1 | a | | see note 1a | X |
| 5.2.1.4.2 | 1.5.2.5.2 | a | | see note 1a | X |
| 5.2.1.4.3 | 1.5.2.5.2 | b | | see note 1a | X |
| 5.2.1.5 | 1.5.2.6 | | Broadcast link management burst | see note 1 | X |
| 5.2.1.5.1 | 1.5.2.6.2 | e | | LME_HO_Ground_Broadcast | E |
| 5.2.1.5.2 | 1.5.2.6.2 | f | | see note 1a | X |
| 5.2.2 | 1.5.3 | | Control (CTRL) parameter formats | see note 1 | X |
| 5.2.2.1 | 1.5.3.1 | | Encoding | see note 1 | X |
| 5.2.2.1.1 | 1.5.3.1 | a | | see note 4 | O |
| 5.2.2.2 | 1.5.3.3 | | General purpose information parameters | see note 1 | X |
| 5.2.2.2.1 | 1.5.3.3 | a | | see note 1a | X |
| 5.2.2.2.2 | 1.5.3.3.1.1 | a | | see note 1a | X |
| 5.2.2.2.3 | 1.5.3.3.1.1 | b | | see note 1a | X |
| 5.2.2.2.4 | 1.5.3.3.1.2 | a | | see note 1a | X |
| 5.2.2.2.5 | 1.5.3.3.1.2 | b | | see note 1a | X |
| 5.2.2.2.6 | 1.5.3.3.2 | a | | see note 1a | X |
| 5.2.2.2.7 | 1.5.3.3.2 | b | | see note 1a | X |
| 5.2.2.2.8 | 1.5.3.3.2 | c | | see note 1a | X |
| 5.2.2.2.9 | 1.5.3.3.2 | d | | see note 1a | X |
| 5.2.2.2.10 | 1.5.3.3.2 | e | | see note 2 | O |

| Requirement reference | Reference in ICAO Doc 9816 [1] | Req | Title | Test case | E/O/X |
|-----------------------|--------------------------------|-----|--|-------------|-------|
| 5.2.2.2.11 | 1.5.3.3.3 | a | | see note 1a | X |
| 5.2.2.2.12 | 1.5.3.3.3 | b | | see note 1a | X |
| 5.2.2.2.13 | 1.5.3.3.3 | c | | see note 2 | O |
| 5.2.2.2.14 | 1.5.3.3.3 | d | | see note 2 | O |
| 5.2.2.2.15 | 1.5.3.3.3 | e | | see note 1a | X |
| 5.2.2.2.16 | 1.5.3.3.3 | f | | see note 1a | X |
| 5.2.2.2.17 | 1.5.3.3.4 | a | | see note 1a | X |
| 5.2.2.2.18 | 1.5.3.3.4 | b | | see note 1a | X |
| 5.2.2.2.19 | 1.5.3.3.4 | c | | see note 1a | X |
| 5.2.2.2.20 | 1.5.3.3.5.1 | a | | see note 1a | X |
| 5.2.2.2.21 | 1.5.3.3.5.1 | b | | see note 1a | X |
| 5.2.2.2.22 | 1.5.3.3.5.1 | c | | see note 1a | X |
| 5.2.2.2.23 | 1.5.3.3.5.2 | a | | see note 1a | X |
| 5.2.2.2.24 | 1.5.3.3.5.2 | b | | see note 1a | X |
| 5.2.2.2.25 | 1.5.3.3.5.2 | c | | see note 1a | X |
| 5.2.2.2.26 | 1.5.3.3.5.2 | d | | see note 1a | X |
| 5.2.2.2.27 | 1.5.3.3.5.2 | e | | see note 1a | X |
| 5.2.2.2.28 | 1.5.3.3.5.3 | a | | see note 2 | O |
| 5.2.2.2.29 | 1.5.3.3.5.4 | a | | see note 2 | O |
| 5.2.2.2.30 | 1.5.3.3.6.1 | a | | see note 1a | X |
| 5.2.2.2.31 | 1.5.3.3.6.1 | b | | see note 1a | X |
| 5.2.2.2.32 | 1.5.3.3.6.1 | c | | see note 1a | X |
| 5.2.2.2.33 | 1.5.3.3.6.2 | a | | see note 1a | X |
| 5.2.2.2.34 | 1.5.3.3.6.1 | d | | see note 1a | X |
| 5.2.2.3 | 1.5.3.4 | | Mobile-initiated information parameters | see note 1 | X |
| 5.2.2.3.a | 1.5.3.4 | a | | see note 2 | O |
| 5.2.2.3.1a | 1.5.3.4.1 | a | | see note 1a | X |
| 5.2.2.3.1b | 1.5.3.4.1 | b | | see note 1a | X |
| 5.2.2.3.2 | 1.5.3.4.1 | c | | see note 1a | X |
| 5.2.2.3.3 | 1.5.3.4.2 | a | | see note 1a | X |
| 5.2.2.3.4 | 1.5.3.4.2 | b | | see note 1a | X |
| 5.2.2.3.5 | 1.5.3.4.2 | c | | see note 1a | X |
| 5.2.2.3.6 | 1.5.3.4.3 | a | | see note 1a | X |
| 5.2.2.3.7 | 1.5.3.4.3 | b | | see note 1a | X |
| 5.2.2.4 | 1.5.3.5 | | Ground-initiated modification parameters | see note 1 | X |
| 5.2.2.4.1 | 1.5.3.5.1 | a | | see note 1a | X |
| 5.2.2.4.2 | 1.5.3.5.1 | b | | see note 1a | X |
| 5.2.2.4.3 | 1.5.3.5.6 | a | | see note 1a | X |
| 5.2.2.4.4 | 1.5.3.5.6 | b | | see note 1a | X |
| 5.2.2.4.5 | 1.5.3.5.6 | c | | see note 1a | X |
| 5.2.2.4.6 | 1.5.3.5.6 | d | | see note 1a | X |
| 5.2.2.4.7 | 1.5.3.5.6 | e | | see note 1a | X |
| 5.2.2.4.8 | 1.5.3.5.8 | a | | see note 1a | X |
| 5.2.2.4.9 | 1.5.3.5.8 | b | | see note 1a | X |
| 5.2.2.4.10 | 1.5.3.5.10 | a | | see note 1a | X |
| 5.2.2.4.11 | 1.5.3.5.10 | b | | see note 1a | X |
| 5.2.2.4.12 | 1.5.3.5.11 | a | | see note 1a | X |
| 5.2.2.4.13 | 1.5.3.5.11 | b | | see note 1a | X |
| 5.2.2.4.14 | 1.5.3.5.12 | a | | see note 1a | X |
| 5.2.2.4.15 | 1.5.3.5.12 | b | | see note 1a | X |
| 5.2.2.4.16 | 1.5.3.5.13 | a | | see note 1a | X |
| 5.2.2.4.17 | 1.5.3.5.13 | b | | see note 1a | X |
| 5.2.2.5 | 1.5.3.6 | | Ground-initiated information parameters | see note 1 | X |
| 5.2.2.5.1 | 1.5.3.6.1 | a | | see note 1a | X |
| 5.2.2.5.2 | 1.5.3.6.1 | b | | see note 1a | X |
| 5.2.2.5.3 | 1.5.3.6.2 | a | | see note 1a | X |
| 5.2.2.5.4 | 1.5.3.6.2 | b | | see note 1a | X |
| 5.2.2.5.5 | 1.5.3.6.2 | c | | see note 1a | X |
| 5.2.2.5.6 | 1.5.3.6.3 | a | | see note 1a | X |
| 5.2.2.5.7 | 1.5.3.6.3 | b | | see note 1a | X |
| 5.2.2.5.8 | 1.5.3.6.4 | a | | see note 1a | X |
| 5.2.2.5.9 | 1.5.3.6.4 | b | | see note 1a | X |

| Requirement reference | Reference in ICAO Doc 9816 [1] | Req | Title | Test case | E/O/X |
|-----------------------|--------------------------------|-----|---|--|-------|
| 5.2.2.5.10 | 1.5.3.6.5 | a | | see note 1a | X |
| 5.2.2.5.11 | 1.5.3.6.5 | b | | see note 1a | X |
| 5.2.2.5.12 | 1.5.3.6.5 | c | | see note 1a | X |
| 5.2.2.5.13 | 1.5.3.6.5 | d | | see note 1a | X |
| 5.2.2.5.14 | 1.5.3.6.5 | e | | see note 1a | X |
| 5.2.3 | 1.5.4 | | LME timers and parameters | see note 1 | X |
| 5.2.3.1 | | | General | see note 1 | X |
| 5.2.3.1.1 | 1.5.4 | a | | see note 2 | O |
| 5.2.3.2 | 1.5.4.2 | | Timer TL1 (maximum link overlap time) | see note 1 | X |
| 5.2.3.2.1 | 1.5.4.2 | a | | LME_TL1_Ground_Init | E |
| 5.2.3.2.2 | 1.5.4.2 | b | | see note 2 | O |
| 5.2.3.2.3 | 1.5.4.2 | c | | LME_TL1_Ground_Init | E |
| 5.2.3.2.4 | 1.5.4.2 | d | | see note 2 | O |
| 5.2.3.2.5 | 1.5.4.2 | e | | see note 2 | O |
| 5.2.3.2.6 | 1.5.4.2 | f | | see note 2 | O |
| 5.2.3.2.7 | 1.5.4.2 | g | | LME_TL1_Mob_Init | E |
| 5.2.3.3 | 1.5.4.3 | | Parameters TL2 (link initialization time) | see note 1 | X |
| 5.2.3.3.1 | 1.5.4.3.1 | a | | LME_HO_TL2 | E |
| 5.2.3.3.2 | 1.5.4.3.2 | a | | LME_HO_TL2 | E |
| 5.2.3.3.3 | 1.5.4.3.2 | b | | see note 2 | O |
| 5.2.3.3.4 | 1.5.4.3.2 | c | | LME_HO_TL2 | E |
| 5.2.3.4 | 1.5.4.4 | | Timer TL4 (leave generation latency) | see note 1 | X |
| 5.2.3.4.1 | 1.5.4.4 | a | | see note 2 | O |
| 5.2.3.4.2 | 1.5.4.4 | b | | see note 2 | O |
| 5.2.3.4.3 | 1.5.4.4 | c | | see note 2 | O |
| 5.2.3.4.4 | 1.5.4.4 | d | | see note 2 | O |
| 5.2.3.4.5 | 1.5.4.4 | e | | see note 2 | O |
| 5.2.4 | 1.5.6 | | CTRL DLPDU types and procedures | see note 1 | X |
| 5.2.4.1 | 1.5.6 | a | | see note 1a | X |
| 5.2.4.2 | 1.5.6 | b | | LME_HO_Receive_Invalid | E |
| 5.2.5 | 1.5.7 | | CTRL transmission procedures | see note 1 | X |
| 5.2.5.a | 1.5.7.1 | | Frequency management procedures | see note 1 | X |
| 5.2.5.a.1 | 1.5.7.1 | a | | see note 1a | X |
| 5.2.5.a.2 | 1.5.7.1.1 | a | | see note 2 | O |
| 5.2.5.a.3 | 1.5.7.1.1 | b | | see note 2 | O |
| 5.2.5.a.4 | 1.5.7.1.1 | c | | see note 2 | O |
| 5.2.5.a.5 | 1.5.7.1.2 | a | | see note 2 | O |
| 5.2.5.a.6 | 1.5.7.1.2 | b | | see note 2 | O |
| 5.2.5.1 | 1.5.7.2 | | Link connectivity procedures | see note 1 | X |
| 5.2.5.1.1 | 1.5.7.2 | a | | see note 1a | X |
| 5.2.5.2 | 1.5.7.3 | | Ground station identification | see note 1 | X |
| 5.2.5.2.3 | 1.5.7.3 | c | | see note 2 | O |
| 5.2.5.2.4 | 1.5.7.3 | d | | see note 2 | O |
| 5.2.5.3 | 1.5.7.4 | | Link establishment | see note 1 | X |
| 5.2.5.3.a | 1.5.7.4 | a | | see note 2 | O |
| 5.2.5.3.b | 1.5.7.4 | b | | see note 2 | O |
| 5.2.5.3.c | 1.5.7.4.1.1 | a | | LME_LE_Station_Choose | E |
| 5.2.5.3.d | 1.5.7.4.1.2 | a | | LME_LE_Station_Choose | E |
| 5.2.5.3.e | 1.5.7.4.1.2 | b | | LME_LE_Station_Choose | E |
| 5.2.5.3.f | 1.5.7.4.1.2 | c | | see note 2 | O |
| 5.2.5.3.6 | 1.5.7.4.3 | b | | LME_LE_Ground_Invalid | E |
| 5.2.5.3.7 | 1.5.7.4.3 | c | | LME_LE_Autotune | E |
| 5.2.5.4 | 1.5.7.5 | | Mobile-initiated handoff | see note 1 | X |
| 5.2.5.4.a | 1.5.7.5 | a | | LME_HO_Mob_Init | E |
| 5.2.5.4.b | 1.5.7.5.1 | a | | see note 2 | O |
| 5.2.5.4.c | 1.5.7.5.1 | b | | LME_HO_TL2 LME_L1_Mob LME_HO_TM2 | E |
| 5.2.5.4.d | 1.5.7.5.1 | c | | LME_HO_TM2 | E |
| 5.2.5.4.e | | | | LME_HO_Mob_Init | E |
| 5.2.5.4.f | 1.5.7.5.2 | a | | see note 2 | O |
| 5.2.5.4.g | 1.5.7.5.3 | a | | see note 4 | O |

| Requirement reference | Reference in ICAO Doc 9816 [1] | Req | Title | Test case | E/O/X |
|-----------------------|--------------------------------|-----|--|--|-------|
| 5.2.5.4.1 | 1.5.7.5.4 | a | | see note 2 | O |
| 5.2.5.4.2 | 1.5.7.5.4 | b | | see note 2 | O |
| 5.2.5.4.7 | 1.5.7.5.6 | a | | see note 2 | O |
| 5.2.5.4.8 | 1.5.7.5.6 | b | | LME_TL1_Mob_Init | E |
| 5.2.5.4.9 | 1.5.7.5.6 | c | | LME_TL1_Mob_Init | E |
| 5.2.5.4.10 | 1.5.7.5.6 | d | | LME_TL1_Mob_Init | E |
| 5.2.5.4.13 | 1.5.7.5.7 | c | | LME_HO_TL2 | E |
| 5.2.5.4.14 | 1.5.7.5.7 | d | | LME_HO_Mob_Init_Auto | E |
| 5.2.5.4.15 | 1.5.7.5.7 | e | | LME_HO_Mob_Init_Grd_Invalid | E |
| 5.2.5.5 | 1.5.7.6 | | Mobile-requested ground-initiated handoff | see note 1 | X |
| 5.2.5.5.a | 1.5.7.6 | a | | see note 2 | O |
| 5.2.5.5.b | 1.5.7.6 | b | | see note 2 | O |
| 5.2.5.5.c | 1.5.7.6.1 | a | | LME_HO_Mob_Req_Grd_Init_A LME_HO_Mob_Req_Grd_Init_B | E |
| 5.2.5.5.d | 1.5.7.6.1 | b | | LME_HO_Mob_Req_Grd_Init_Aff | E |
| 5.2.5.5.5 | 1.5.7.6.4 | a | | see note 4 | O |
| 5.2.5.6 | 1.5.7.7 | | Ground-initiated handoff | see note 1 | X |
| 5.2.5.6.5 | 1.5.7.7.1 | c | | see note 2 | O |
| 5.2.5.6.5a | 1.5.7.7.2 | a | | LME_HO_Ground_Init LME_HO_Ground_Init_List | E |
| 5.2.5.6.5b | 1.5.7.7.3 | a | | LME_TL1_Ground_Init | E |
| 5.2.5.6.7 | 1.5.7.7.3 | c | | LME_TL1_Ground_Init | E |
| 5.2.5.6.7a | 1.5.7.7.4 | a | | LME_HO_Ground_Init_Invalid | E |
| 5.2.5.7 | 1.5.7.8 | | Ground-requested mobile-initiated handoff | see note 1 | X |
| 5.2.5.7.5a | 1.5.7.8.2 | a | | LME_HO_Gr_Req_Mob_Init | E |
| 5.2.5.7.5b | 1.5.7.8.3 | a | | LME_HO_Gr_Req_Mob_Init_Invalid | E |
| 5.2.5.7.5c | 1.5.7.8.3 | b | | LME_HO_Gr_Req_Mob_Init_Auto LME_HO_Gr_Req_Mob_Init_Auto_Retrans | E |
| 5.2.5.7.5d | 1.5.7.8.3 | c | | LME_HO_Gr_Req_Mob_Init | E |
| 5.2.5.8 | 1.5.7.9 | | Ground-requested broadcast handoff | see note 1 | X |
| 5.2.5.8.4 | 1.5.7.9.2 | a | | LME_HO_Ground_Broadcast | E |
| 5.2.5.8.5 | 1.5.7.9.2 | b | | LME_HO_Ground_Broadcast | E |
| 5.2.5.8.6 | 1.5.7.9.2 | c | | LME_HO_Ground_Broadcast | E |
| 5.2.5.8.7 | 1.5.7.9.3 | a | | LME_HO_GND_Broadcast_Invalid_A | E |
| 5.2.5.8.8 | 1.5.7.9.3 | b | | LME_HO_GND_Broadcast_Invalid_B | E |
| 5.2.5.8.9 | 1.5.7.9.3 | c | | LME_HO_GND_Broadcast_Invalid_C LME_HO_GND_Broadcast_Invalid_D | E |
| 5.2.5.9 | 1.5.7.10 | | Ground-commanded autotune | see note 1 | X |
| 5.2.5.9.2 | 1.5.7.10.2 | a | | LME_HO_Gr_Req_Mob_Init_Auto | E |
| 5.2.5.9.3 | 1.5.7.10.3 | a | | LME_HO_Mob_Init_Auto LME_LE_Autotune | E |
| 5.2.6 | 2 | | VDL Mode 4 Mobile SubNetwork Dependent Convergence Function (SND CF) | see note 1 | X |
| 5.2.6.1 | | | Frame Mode SND CF | see note 1 | X |
| 5.2.6.1.1 | 2 | a | | see note 2 | O |
| 5.2.6.1.2 | 2 | b | | see note 2 | O |
| 5.2.6.1.3 | 2 | c | | see note 2 | O |
| 5.2.6.1.4 | 2 | d | | see note 2 | O |
| 5.3 | 1.3 | | Additional VSS requirements | see note 1 | X |
| | | | | | |
| | | | | | |
| 5.3.1 | 1.3.15 | | Information transfer request protocol specification | see note 1 | X |
| 5.3.1.1 | 1.3.15.2 | | Information transfer request parameters | see note 1 | X |
| 5.3.1.1.1 | 1.3.15.2 | a | | see note 1a | X |
| 5.3.1.1.2 | 1.3.15.2 | b | | see note 2 | E |

| Requirement reference | Reference in ICAO Doc 9816 [1] | Req | Title | Test case | E/O/X |
|-----------------------|--------------------------------|-----|---|--------------------------|-------|
| 5.3.1.1.3 | 1.3.15.2.1 | a | | see note 1a | X |
| 5.3.1.1.4 | 1.3.15.2.2 | a | | see note 1a | X |
| 5.3.1.1.5 | 1.3.15.2.3 | a | | see note 1a | X |
| 5.3.1.1.6 | 1.3.15.2.4 | a | | see note 1a | X |
| 5.3.1.1.7 | 1.3.15.2.5 | a | | see note 1a | X |
| 5.3.1.2 | 1.3.15.4 | | Information transfer request transmission procedures | see note 1 | X |
| 5.3.1.2.1 | 1.3.15.4.1 | a | | DLS_Initialize_Send_Long | E |
| 5.3.1.2.2 | 1.3.15.4.1 | b | | see note 1a | X |
| 5.3.1.2.3 | 1.3.15.4.2 | a | | DLS_Initialize_Send_Long | E |
| 5.3.1.2.4 | 1.3.15.4.2 | b | | see note 1a | X |
| 5.3.1.2.5 | 1.3.15.4.3 | a | | DLS_Initialize_Send_Long | E |
| 5.3.1.2.6 | 1.3.15.4.4 | a | | DLS_Initialize_Send_Long | E |
| 5.3.1.2.7 | 1.3.15.4.4 | b | | see note 1a | X |
| 5.3.1.2.8 | 1.3.15.4.5 | a | | see note 2 | O |
| 5.3.1.3 | 1.3.15.5 | | Information transfer request acknowledgement procedures | see note 1 | X |
| 5.3.1.3.1 | 1.3.15.5 | a | | DLS_Initialize_Send_Long | E |

List of tables

| | |
|---|----|
| Table 4.1: DLPDU contents | 15 |
| Table 5.1: DLS state variables | 23 |
| Table 5.2: Normal unicast DLS burst format | 24 |
| Table 5.3: Data DLPDU encoding | 24 |
| Table 5.4: Two-Octet DLPDUs encoding | 25 |
| Table 5.5: Single octet DLPDUs encoding | 25 |
| Table 5.5a: Interpretation of negotiation subfield | 26 |
| Table 5.6: Compressed combined RTS/INFO (type 1) burst format..... | 27 |
| Table 5.7: Compressed combined RTS/INFO (type 2) burst format..... | 27 |
| Table 5.8: Data link service system parameters | 28 |
| Table 5.9: Re-transmission parameters | 29 |
| Table 5.10: Short transmission INFO DLPDU parameters | 32 |
| Table 5.11: Long transmission RTS DLPDU parameters | 32 |
| Table 5.12: CTS DLPDU parameters..... | 33 |
| Table 5.13: CTRL DLPDU format..... | 39 |
| Table 5.14: Parameter set identifier encoding | 40 |
| Table 5.15: Connection management parameter | 41 |
| Table 5.16: Connection management parameter values | 41 |
| Table 5.17: Abbreviated CTRL names | 41 |
| Table 5.18: CTRL sequencing parameter..... | 41 |
| Table 5.19: Protocol options parameter..... | 42 |
| Table 5.20: Protocol options parameter values | 42 |
| Table 5.21: LCR cause parameter | 43 |
| Table 5.22: Cause code table..... | 43 |
| Table 5.23: UCTRL_DM parameter | 44 |
| Table 5.24: Modulation support list encoding..... | 44 |
| Table 5.25: Modulation encoding..... | 44 |
| Table 5.26: Acceptable alternate ground station parameter..... | 45 |
| Table 5.27: Destination airport parameter | 45 |
| Table 5.28: Data link service parameter encoding | 46 |
| Table 5.29: Autotune parameter encoding | 46 |
| Table 5.30: Replacement ground station list parameter encoding..... | 46 |
| Table 5.31: Re-transmission parameter encoding | 47 |

| | |
|---|----|
| Table 5.32: Timer TL1 parameter encoding | 47 |
| Table 5.33: Ground station address filter parameter | 48 |
| Table 5.34: Broadcast connection parameter encoding | 48 |
| Table 5.35: Airport coverage indication parameter encoding | 48 |
| Table 5.36: Nearest airport parameter encoding | 49 |
| Table 5.37: ATN router NETs parameter encoding | 49 |
| Table 5.38: Ground-based system mask parameter encoding | 50 |
| Table 5.39: Frequency support list encoding | 50 |
| Table 5.40: Management entity system parameters | 50 |
| Table 5.41: CTRL parameters | 52 |
| Table 5.41 (continued): CTRL parameters | 53 |
| Table 5.41 (concluded): CTRL parameters | 54 |
| Table 5.42: Information transfer request VSS system parameters | 61 |
| Table 7.1: Protocol test-suite overview | 64 |
| Table 7.2: Subfield mnemonics | 67 |
| Table 7.3: Special characters used in the subfield definitions | 68 |
| Table 7.4: CTRL_RTS_a (CRa): CTRL_RTS DLPDU with unicast reservation | 69 |
| Table 7.5: INFO_RTS_a (IRa): INFO_RTS DLPDU with unicast reservation | 69 |
| Table 7.6: INFO_CTS_a (ICa): INFO_CTS DLPDU with information transfer reservation | 70 |
| Table 7.7: CTRL_CTS_a (CCa): CTRL_CTS DLPDU with information transfer reservation | 71 |
| Table 7.8: UDATA_CTS_a (UDCa): UDATA_CTS DLPDU with unicast reservation | 71 |
| Table 7.9: INFO_a (IFa): INFO DLPDU with unicast reservation | 72 |
| Table 7.10: CTRL_a (CTa): CTRL DLPDU with unicast reservation | 72 |
| Table 7.11: CTRL_b (CTb): CTRL DLPDU with response reservation | 73 |
| Table 7.12: CTRL_ACK_a (CAa): CTRL_ACK DLPDU with response reservation | 73 |
| Table 7.13: INFO_b (IFb): INFO DLPDU with response reservation | 74 |
| Table 7.14: INFO_ACK_a (IAa): INFO_ACK DLPDU with response reservation | 74 |
| Table 7.15: INFO_RTS_SZOM_a (IRSZa): INFO_RTS DLPDU and SZOM DLPDU combined burst with unicast reservation | 74 |
| Table 7.16: INFO_CTS_SZOM_a (ICSZa): INFO_CTS DLPDU and SZOM DLPDU combined burst with response reservation | 75 |
| Table 7.17: INFO_ACK_SZOM_a (IASZa): INFO_ACK DLPDU and SZOM DLPDU combined burst with response reservation | 76 |
| Table 7.18: INFO_SZOM_a (IFSZa): INFO DLPDU and SZOM DLPDU combined burst with unicast reservation | 76 |
| Table 7.19: INFO_RTS_INFO_b (IRIFb): INFO_RTS and INFO DLPDU combined burst with response reservation | 77 |
| Table 7.20: INFO_ACK_INFO_CTS_a (IAICa): INFO_ACK DLPDU and INFO_CTS DLPDU combined burst with information transfer reservation | 77 |

| | |
|--|-----|
| Table 7.21: DM_DISC_a (DMDCa): DM/DISC DLPDU with response reservation..... | 78 |
| Table 7.22: UINFO_a (UIa): UINFO DLPDU with response reservation with address type field equal to 7 | 78 |
| Table 7.23: UCTRL_a (UCa): UCTRL DLPDU with response reservation with address type field equal to 7 | 78 |
| Table 7.24: UCTRL_b (UCb): UCTRL DLPDU with connection management, control sequencing, ground station address filter, broadcast connection, ATN router nets and response reservation with address type field equal to 7.79 | |
| Table 7.25: GEN_RESP_b (GRb): General response burst with unicast reservation | 80 |
| Table 7.26: INFO_ACK_INFO_a (IAIFa): INFO_ACK and INFO DLPDU combined burst with unicast reservation .. | 80 |
| Table 7.27: INFO_ACK_INFO_RTS_a (IAIRa): INFO_ACK and INFO_RTS DLPDU combined burst with unicast reservation | 81 |
| Table 7.28: CTRL_c (CTc): CTRL DLPDU with connection management, control sequencing, and unicast reservation | 81 |
| Table 7.29: CTRL_d (CTd): CTRL DLPDU with connection management, control sequencing, and response reservation | 82 |
| Table 7.30: CTRL_e (CTe): Invalid CTRL DLPDU with connection management, control sequencing, and response reservation | 82 |
| Table 7.31: CTRL_g (CTg): Invalid CTRL DLPDU with connection management, control sequencing, and unicast reservation | 83 |
| Table 7.32: CTRL_h (CTh): CTRL DLPDU with connection management, control sequencing, ATN router NETs, ground-based system mask, and response reservation | 84 |
| Table 7.33: CTRL_i (CTi): CTRL DLPDU with connection management, control sequencing, ATN router NETs, ground-based system mask, replacement ground station list, and response reservation | 85 |
| Table 7.34: CTRL_k (CTk): CTRL DLPDU with connection management, control sequencing, autotune, replacement ground station list, and response reservation..... | 86 |
| Table 7.35: CTRL_ACK_CTRL_RTS_a (CACRa): CTRL_ACK and CTRL_RTS DLPDU combined burst with unicast reservation | 87 |
| Table 7.36: CTRL_l (CTl): CTRL DLPDU with connection management, control sequencing, replacement ground station list, and response reservation | 88 |
| Table 7.37: CTRL_m (CTm): CTRL DLPDU with connection management, control sequencing, ground-based system mask, and response reservation | 89 |
| Table 7.38: CTRL_n (CTn): Invalid CTRL DLPDU with connection management, control sequencing, ground-based system mask and unicast reservation..... | 90 |
| Table 7.39: CTRL_o (CTo): CTRL DLPDU with connection management, control sequencing, autotune, replacement ground station list, ground-based system mask and response reservation..... | 91 |
| Table 7.40: CTRL_q (CTq): CTRL DLPDU with connection management, control sequencing, protocol options and response reservation | 92 |
| Table 7.41: CTRL_r (CTr): Invalid CTRL DLPDU with connection management, control sequencing, ATN router NETs, ground-based system mask and response reservation | 93 |
| Table 7.42: CTRL_s (CTs): CTRL DLPDU with connection management, control sequencing, LCR cause, and response reservation | 94 |
| Table 7.43: CTRL_t (CTt): CTRL DLPDU with connection management, control sequencing, ground-based system mask, random access and response reservation..... | 95 |
| Table A.1: VDL Mode 4 requirements according to ICAO VDL4 Technical Manual [1]..... | 173 |

History

| Document history | | |
|------------------|---------------|---|
| V1.1.1 | July 2005 | Publication |
| V1.2.1 | December 2006 | Publication |
| V1.2.2 | December 2010 | Publication |
| V1.3.0 | December 2014 | EN Approval Procedure AP 20150404: 2014-12-05 to 2015-04-06 |
| V1.3.1 | April 2015 | Publication |