

ETSI EN 302 842-3 V1.1.1 (2005-07)

European Standard (Telecommunications series)

**Electromagnetic compatibility
and Radio spectrum Matters (ERM);
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**



Reference

DEN/ERM-TG25-016-5-3

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

Individual copies of the present document can be downloaded from:

<http://www.etsi.org>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the 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:

http://portal.etsi.org/chaicor/ETSI_support.asp

Copyright Notification

No part may be reproduced except as authorized by written permission.
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2005.
All rights reserved.

DECTTM, **PLUGTESTS**TM and **UMTS**TM are Trade Marks of ETSI registered for the benefit of its Members.
TIPHONTM and the **TIPHON logo** are Trade Marks currently being registered by ETSI for the benefit of its Members.
3GPPTM is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

Contents

| | |
|---|----|
| Intellectual Property Rights | 5 |
| Foreword..... | 5 |
| Introduction | 6 |
| 1 Scope | 7 |
| 2 References | 8 |
| 3 Definitions and abbreviations..... | 9 |
| 3.1 Definitions | 9 |
| 3.1.1 Basic reference model definitions..... | 9 |
| 3.1.2 Service conventions definitions | 9 |
| 3.1.3 General definitions..... | 9 |
| 3.1.4 Definition of bit order..... | 10 |
| 3.2 Abbreviations | 10 |
| 4 General description of VDL Mode 4 broadcast services..... | 11 |
| 4.1 General | 11 |
| 4.2 Automatic Dependant Surveillance - Broadcast (ADS-B) | 11 |
| 4.3 Traffic Information Service - Broadcast (TIS-B) | 12 |
| 4.4 Flight Information Service - Broadcast (FIS-B)..... | 13 |
| 4.5 GNSS Augmentation Service - Broadcast (GNS-B) | 14 |
| 5 Minimum performance specification under standard test conditions..... | 15 |
| 5.1 Requirements for ADS-B | 15 |
| 5.1.1 Information Field Formats | 15 |
| 5.1.2 ADS-B Request | 25 |
| 5.1.2.1 ADS-B request format..... | 25 |
| 5.1.2.2 Sleep Mode | 25 |
| 5.1.2.3 Automatic Selection of Variable Information Fields | 25 |
| 5.1.3 Default ADS-B Reporting | 27 |
| 5.1.4 ADS-B Procedures..... | 27 |
| 5.1.5 CTRL Parameters | 29 |
| 5.1.5.1 Directory of Service (DOS) message | 29 |
| 5.1.5.2 Channel Management Parameter | 29 |
| 5.1.6 Definitions for Compact Position Reporting | 39 |
| 5.1.6.1 Information Field Offset Encoding | 39 |
| 5.1.6.2 Information Field Offset Decoding..... | 41 |
| 5.1.6.3 Patch ID Encoding | 42 |
| 5.1.6.4 Patch Id Decoding..... | 43 |
| 5.1.7 Encoding of UDATA (udid) | 43 |
| 5.2 Requirements for TIS-B | 44 |
| 5.2.1 Traffic Information Volume (TIV) | 44 |
| 5.2.2 Void | 44 |
| 5.2.3 Message format..... | 44 |
| 5.2.4 Management message | 45 |
| 5.2.5 Aircraft target messages (airborne TIV)..... | 48 |
| 5.2.6 Aircraft target messages (ground TIV) | 49 |
| 5.2.7 Ground vehicle target messages (ground TIV) | 51 |
| 5.2.8 TIS-B offset encoding..... | 51 |
| 5.3 Requirements for FIS-B | 54 |
| 5.3.1 Message format for received FIS-B messages | 54 |
| 5.3.2 Meteorological Aerodrome Report (METAR) message | 55 |
| 5.3.3 Special Observations and Reports (SPECI) message..... | 62 |
| 5.3.4 Automatic Terminal Information Service (ATIS) message | 63 |
| 5.3.5 Runway Condition (RCN) message..... | 65 |
| 5.3.6 Significant Meteorological Information (SIGMET) message..... | 69 |
| 5.3.7 Temporary Segregated Areas (TSA) message | 70 |

| | | |
|-------------------------------|--|------------|
| 5.3.8 | FIS-B report request message | 73 |
| 5.3.9 | Void | 74 |
| 5.3.9a | Airborne system functions | 74 |
| 5.3.9a.1 | Transmission of requests | 74 |
| 5.3.9a.2 | Differentiation between reports..... | 74 |
| 5.4 | Requirements for GNSS Augmentation Service Broadcast (GNS-B) message | 75 |
| 5.4.1 | Message format..... | 75 |
| 5.4.2 | Message Type 1 | 76 |
| 5.4.3 | Message Type 2 | 79 |
| 5.4.4 | Message Type 4 | 83 |
| 5.4.5 | CRC Calculation..... | 89 |
| 5.4.5.1 | GNS-B CRC..... | 89 |
| 5.4.5.2 | FAS CRC | 89 |
| 5.4.5.3 | Ephemeris CRC..... | 90 |
| 6 | General design requirements | 90 |
| 7 | Protocol test procedures | 90 |
| 7.1 | General | 90 |
| 7.2 | Required test rig | 91 |
| 7.3 | Protocol test-suite description methodology | 91 |
| 7.4 | Detailed protocol test procedures | 91 |
| 7.4.1 | Test-suite overview..... | 91 |
| 7.4.2 | Declarations | 94 |
| 7.4.3 | Constraints | 94 |
| 7.4.3.1 | Abbreviations | 94 |
| 7.4.3.1.1 | Subfield mnemonics | 94 |
| 7.4.3.1.2 | Station addresses and positions | 98 |
| 7.4.3.1.3 | Tables of values for use in CPR test cases | 99 |
| 7.4.3.1.4 | Tables of values for use in content checking test cases | 108 |
| 7.4.3.1.5 | VDL4 Burst formats | 144 |
| 7.4.3.2 | Test cases | 182 |
| 7.4.3.2.1 | Test case macros | 183 |
| 7.4.3.2.2 | Test case descriptions | 184 |
| Annex A (informative): | Cross reference matrix | 259 |
| Annex B (informative): | Bibliography | 268 |
| History | | 275 |

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://webapp.etsi.org/IPR/home.asp>).

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 (Telecommunications series) has been produced by ETSI Technical Committee Electromagnetic compatibility and Radio spectrum Matters (ERM).

The present document is part 3 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, EN 301 842 [9] parts 1 to 4, 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 (see bibliography) which are a sub set of EN 302 842-1, 2, 3 and 4. EN 302 842-1, 2, 3 and 4 complies with the requirements of CEC Mandate M/318.

| National transposition dates | |
|--|-----------------|
| Date of adoption of this EN: | 22 July 2005 |
| Date of latest announcement of this EN (doa): | 31 October 2005 |
| Date of latest publication of new National Standard or endorsement of this EN (dop/e): | 30 April 2006 |
| Date of withdrawal of any conflicting National Standard (dow): | 30 April 2006 |

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 118,000 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 EN 302 842-1 [4] and EN 302 842-2 [5].

NOTE: The present document has been produced with a view to maintaining consistency of numbering with the equivalent standard for ground equipment (EN 301 842 all parts [9]). 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 EN 301 842 [9]. This has led to a non-standard form of numbering for new requirements in some places.

1 Scope

The present document provides part 3 of 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 118,000 MHz to 136,975 MHz as defined in ICAO VDL SARPs [2].

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 may also cover any 25 kHz channel from 112,000 MHz to 117,975 MHz and the receiver(s) may cover any 25 kHz channel from 108,000 MHz to 117,975 MHz.

The present document applies to "aeronautical mobile (airborne and in some cases 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 EN 301 842 [9].

A description of the scope of the VDL Mode 4 system is provided in part 2 of these technical specifications (see EN 302 842-2 [5] clause 1).

EN 302 842-1 [4] deals with tests of the physical layer. EN 302 842-2 [5] deals with tests of the link layer sufficient to support broadcast functionality including requirements and tests sufficient to recognize and respond to transmissions associated with point-to-point communication. The present document provides technical specifications for a VDL Mode 4 mobile transceiver supporting a full Automatic Dependent Surveillance - Broadcast (ADS-B) capability and, optionally, the additional functionality of either, or a combination of, the following services:

- Traffic Information Service - Broadcast (TIS-B);
- Flight Information Service - Broadcast (FIS-B);
- GNSS Augmentation Service - Broadcast (GNS-B).

The TIS-B, FIS-B, or GNS-B reception processing functionality in the airborne equipment is expected to be provided by a TIS-B, FIS-B or GNS-B processor, which could be contained within the VDL Mode 4 transceiver, but could also be housed in a separate physical unit. Therefore to support TIS-B, FIS-B or GNS-B message reception, the minimum functionality demanded of a basic VDL Mode 4 airborne transceiver unit (i.e. one that does not have a TIS-B, FIS-B or GNS-B processor housed within it) is to pass to the TIS-B, FIS-B or GNS-B processor, all TIS-B, FIS-B or GNS-B messages received.

As the measured values of equipment performance may be a function of the method of measurement, standard test conditions and methods of test are recommended in the present document.

The present document is organized as follows:

- references, definitions, abbreviations and symbols are provided in clauses 2 and 3;
- clause 4 describes the VDL Mode 4 mobile station functionality to support ADS-B, TIS-B, FIS-B and GNS-B;
- clause 5 provides performance specifications for the VDL Mode 4 mobile station supporting ADS-B, TIS-B, FIS-B and GNS-B Services;
- clause 6 provides general design requirements;
- clause 7 provides protocol tests which emphasis the ADS-B, TIS-B, FIS-B and GNS-B functions of the system;
- annex A provides a detailed cross-reference to the relevant requirements contained in ICAO VDL4 Technical Manual [1];
- annex B provides a bibliography;

- a document history.

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 functionality to support ADS-B, FIS-B, TIS-B and GNS-B focusing on the system 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

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

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

- [1] ICAO 9816: "Manual on VHF Digital Link (VDL) Mode 4 - Part 2: Detailed Technical Specifications". First Edition 2004.
- [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".
- [3] RTCA DO-242A: "Minimum Aviation System Performance Standards for Automatic Dependent Surveillance Broadcast (ADS-B)".
- [4] ETSI EN 302 842-1: "Electromagnetic compatibility and Radio spectrum Matters (ERM); 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".
- [5] ETSI EN 302 842-2: "Electromagnetic compatibility and Radio spectrum Matters (ERM); 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".
- [6] ETSI EN 302 842-4: "Electromagnetic compatibility and Radio spectrum Matters (ERM); 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".

- [7] Amendments 76 and 77 to Volume I of Annex 10 to the Convention on International Civil Aviation, International Civil Aviation Organization: Appendix B-B2, 3.6 "Ground-based augmentation system (GBAS)".

NOTE: <http://www.icao.int>.

- [8] EUROCAE ED-114: "Minimum Operational Performance Specification for Global Navigation Satellite Ground Based Augmentation System Ground Equipment to Support Category I Operations".

NOTE: <http://www.eurocae.org>.

- [9] ETSI EN 301 842 (all parts): "Electromagnetic compatibility and Radio spectrum Matters (ERM); VHF air-ground Data Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for ground-based equipment".

3 Definitions and abbreviations

3.1 Definitions

3.1.1 Basic reference model definitions

See EN 302 842-2 [5] clause 3.1.1.

3.1.2 Service conventions definitions

See EN 302 842-2 [5] clause 3.1.2.

3.1.3 General definitions

For the purposes of the present document, the terms and definitions given in EN 302 842-1 [4] clause 3.1.3, EN 302 842-2 [5] clause 3.1.3 and the following apply:

Automatic Dependent Surveillance-Broadcast (ADS-B): surveillance application transmitting parameters, such as position, track, ground speed and time via a broadcast mode data link for use by any air and ground users requiring it

NOTE: ADS-B is a surveillance service based on aircraft self-determination of position/velocity/time and automatic, periodic, broadcast of this information along with auxiliary data such as aircraft identity (ID), intent information and communications control parameters, etc. ADS-B is intended to support multiple high-level applications and associated services such as cockpit display of traffic information, traffic alert and collision avoidance functionality, enhanced traffic management in the air and on the ground, search and rescue support and others.

Automatic Terminal Information Service (ATIS): report generated by a ground station that includes weather conditions, operating procedures, runways and approaches in use, and any other information that may affect the departure, runway and landing phase of flight

Flight Information Service - Broadcast (FIS-B): uplink broadcast application providing local airborne traffic with information

GNSS Augmentation Service - Broadcast (GNS-B): uplink broadcast application providing aircraft with GNSS augmentation and integrity data

METEorological Aerodrome Report (METAR): report generated by a ground station that broadcasts meteorological conditions at aerodromes, officially recoded and communicated at regular intervals

SIGNificant METeorological Information (SIGMET): report generated by a ground station that broadcasts information about weather phenomena that may have an impact on aircraft at subsonic, transonic and supersonic cruising levels including thunderstorms, cyclones, turbulence and icing

SPECIAL Observations and Reports (SPECI): report generated by a ground station that is issued when meteorological conditions change sufficiently to affect aviation operations

NOTE: The SPECI report contains the same information as a METAR.

Traffic Information Service - Broadcast (TIS-B): uplink surveillance service that derives traffic information from one or more ground surveillance sources and broadcasts that information to suitably equipped aircraft or surface vehicles

Traffic Information Volume (TIV): volume of airspace for which surveillance information is provided for all targets

NOTE: Inside the TIV, a pilot knows he can rely on the surveillance picture presented to him; outside the TIV, ADS-B reports may be received but TIS-B reports may not be available.

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 shall be indicated by subscript numbers. Bit 1 shall indicate the least significant bit; and
- b) bits shall be transmitted octet by octet, starting with the first octet in each table, and within each octet the rightmost bit (as shown in the tables) shall be transmitted first.

3.2 Abbreviations

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

| | |
|---------|---|
| ADS-B | Automatic Dependent Surveillance - Broadcast |
| ATIS | Automatic Terminal Information Service |
| CCI | Co-Channel Interference |
| CPR | Compact Position Reporting |
| CRC | Cyclic Redundancy Code |
| dB | decibel |
| DLPDU | Data Link Protocol Data Unit |
| DLS | Data Link Service |
| DOS | Directory Of Services |
| EUROCAE | EUROpean Organization for Civil Aviation Equipment |
| FAS | Final Approach Segment |
| FIS-B | Flight Information Service - Broadcast |
| FPAP | Flight Path Alignment Point |
| FTP | Fictitious Threshold Point |
| GBAS | Ground-Based Augmentation System |
| GCID | GNS-B Continuity / Integrity Designator |
| GFSK | Gaussian Filtered Frequency Shift Keying |
| GLONASS | GLOBAL NAVigation Satellite System (Russian system) |
| GNS-B | GNSS augmentation Service - Broadcast |
| GNSS | Global Navigation Satellite System |
| GPA | Glide Path Angle |
| GPS | Global Positioning System |
| GRAS | Ground-based Regional Augmentation System |
| GSC | Global Signalling Channel |
| hex | hexadecimal |
| IA-5 | International Alphabet-5 |
| ICAO | International Civil Aviation Organization |
| ID | Identity |
| INFO | INFORMATION (DLPDU) |
| LTP | Landing Threshold Point |
| MASPS | Minimum Aviation System Performance Standards |
| MEDUP | MEDiterranean Update Programme |
| METAR | METEorological Aerodrome Report |
| MOPS | Minimum Operational Performance Specification |

| | |
|--------|---|
| NEAN | North European ADS-B Network |
| NM | Nautical Mile |
| NUP | NEAN Update Programme |
| PCO | Point of Control and Observation |
| PRN | Pseudo Random Noise |
| RCN | Runway Condition |
| RF | Radio Frequency |
| SARPs | Standards And Recommended Practices |
| SBAS | Space-Based Augmentation System |
| SIGMET | SIGNificant METeorological event |
| SPECI | SPECIAL observations and reports |
| TCH | Threshold Crossing Height |
| TCP | Trajectory Change Point |
| TIS-B | Traffic Information Service - Broadcast |
| TIV | Traffic Information Volume |
| TV | Transmission Volume |
| UTC | Universal Time Coordinated |
| VDL | VHF Digital Link |
| VHF | Very High Frequency |
| VSS | VDL Mode 4 Specific Services |

4 General description of VDL Mode 4 broadcast services

4.1 General

A description of VDL Mode 4 is provided in clause 4.1 of EN 302 842-2 [5]. This clause provides a description of the assumptions made in the derivation of the requirements for the VDL Mode 4 mobile station.

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 that apply explicitly only to ground stations. A number of other requirements will also not apply because of the assumed services provided by the mobile station. The assumed services provided by the mobile station and the impact on the requirements are summarized in the rest of clause 4.

The scope of the present document is for a mobile station supporting broadcast applications. Hence the ability to support point-to-point communication is not included in the present document. Those requirements are presented in EN 302 842-4 [6].

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

4.2 Automatic Dependant Surveillance - Broadcast (ADS-B)

Automatic Dependent Surveillance - Broadcast (ADS-B) is a surveillance application in which aircraft, vehicles and ground stations broadcast their identity, position, velocity, time, intent and other information, enabling other aircraft, vehicles and ground stations to develop a surveillance picture.

ADS-B relies on the regular and frequent transmission of position reports via a broadcast data link. The position reports are sent periodically by the aircraft with no intervention from the ground. Position reports may be received by any recipient in range of the transmitting aircraft. Recipients may be other aircraft, ground vehicles, or fixed ground sites.

ADS-B offers data delivery from aircraft-to-aircraft or from aircraft-to-ground. Transmitting data directly from air-to-air means that there is no need for a ground infrastructure to be present for airborne surveillance to be performed. By using position reports received from surrounding aircraft, a traffic surveillance picture can be generated in the cockpits of all aircraft. This potentially allows new applications or new manoeuvres to be performed by pilots.

The transmitting aircraft does not know which, if any, recipients are receiving and processing the position reports. Unlike a point-to-point link, position reports are not acknowledged. The concept with ADS-B is that position reports are transmitted so frequently that the loss of a small number of position reports is not operationally significant.

ADS-B messages are broadcast and received by appropriately equipped participant subsystems. ADS-B subsystems include aircraft, vehicle, and ground subsystems. The capabilities of participant subsystems will vary based upon class of equipment. The ADS-B aircraft subsystem interacts with other onboard systems such as pilot display equipment and the aircraft navigation system. The ADS-B ground subsystem interacts with other ground systems such as ATM applications and controller display processing equipment.

If received by a data acquisition unit, the position report will be processed with other surveillance data and may be forwarded to a controller/pilot display.

Airborne VDL Mode 4 stations supporting full ADS-B capability:

- Receive and process ADS-B reports from other aircraft, vehicles and ground stations, passing the received data to a surveillance server.
- Transmit ADS-B reports at the required regular intervals (or more frequently) and including required and possibly some optional information.

Figure 4.1 illustrates the context for the airborne station supporting ADS-B functions.

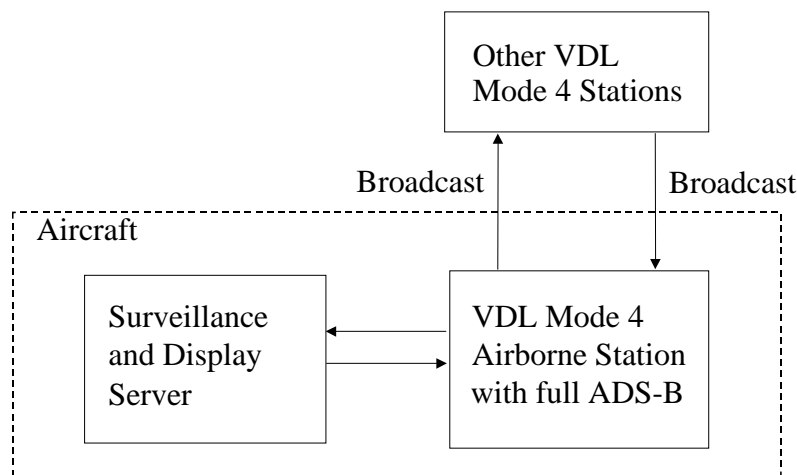


Figure 4.1: Airborne station supporting ADS-B functions

A VDL Mode 4 airborne station supporting ADS-B receives broadcast ADS-B reports and passes on the data to a surveillance processor. A VDL Mode 4 airborne station supporting ADS-B may also receive data for transmission and instructions for transmission rates from the surveillance processor.

The requirements for ADS-B are taken from the ICAO VDL 4 Technical Manual [1].

4.3 Traffic Information Service - Broadcast (TIS-B)

Traffic Information Service - Broadcast (TIS-B) is a surveillance service that derives traffic information from one or more ground surveillance sources and broadcasts that information to suitably equipped aircraft or surface vehicles.

The purpose of TIS-B is to complement the surveillance information provided from ADS-B aircraft to ensure that a full surveillance picture is available to airborne systems.

The TIS-B service is provided within an area known as the Traffic Information Volume (TIV), which is the volume of airspace for which surveillance information is provided for all targets. Inside the TIV, a pilot knows he can rely on the surveillance picture presented to him; outside the TIV, ADS-B reports may be received but TIS-B reports may not be available.

A TIS-B service either provides a "full surveillance picture" or a "gap filler service". For the full surveillance picture, information is provided on all targets. The "gap filler service" information is only provided for targets which do not support ADS-B via VDL Mode 4.

A VDL Mode 4 airborne station supporting TIS-B receives two main sorts of message:

- a) Management Messages: which contain information about the TIS-B service and the TIV.
- b) Target Messages: which contain information about aircraft or ground vehicle targets.

Figure 4.2 illustrates the context for the airborne station supporting TIS-B.

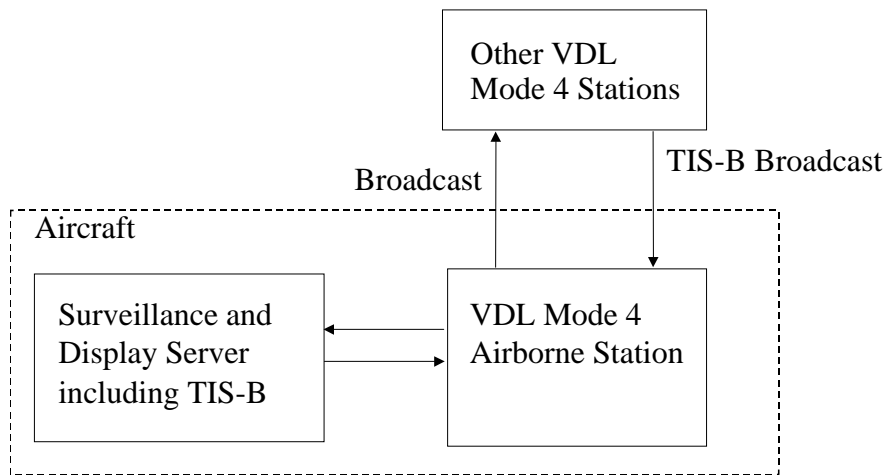


Figure 4.2: Airborne station supporting TIS-B functions

A VDL Mode 4 airborne station supporting TIS-B receives broadcast TIS-B reports and passes on the data to a surveillance processor that includes TIS-B processing capability. The present document therefore includes requirements that apply to the part of the airborne equipment involved in TIS-B data processing, but this is not necessarily contained within the VDL Mode 4 airborne station.

The requirements for TIS-B are taken from the TIS-B Service Description produced by the Mediterranean UpDate and Nean Update programmes (MEDUP and NUP). Eurocontrol TIS-B Requirements have been taken into account in the development of the present requirements.

4.4 Flight Information Service - Broadcast (FIS-B)

Flight Information Service - Broadcast (FIS-B) is an uplink broadcast application providing local traffic with information.

The following message types are supported:

- a) Meteorological Aerodrome Report (METAR): a report indicating meteorological conditions at aerodromes, officially recorded and communicated at regular intervals. Intermediate observations are also recorded and communicated when meteorological conditions change sufficiently to affect aviation operations (see SPECI).
- b) Automatic Terminal Information Service (ATIS): a report indicating weather conditions, operating procedures, runways and approaches in use, and any other information that may affect the departure, runway and landing phase of flight.
- c) Runway Condition (RCN): a report including information about the current runway visual range conditions and any other runway condition that may affect the departure, runway and landing phases of flight.
- d) Significant Meteorological Information (SIGMET): a report broadcasting information about weather phenomena which may have an impact on aircraft at subsonic, transonic and supersonic cruising levels including thunderstorms, cyclones, turbulence and icing.
- e) Special Observations and Reports (SPECI): a report generated when meteorological conditions change sufficiently to affect aviation operations. The SPECI report contains the same information as a METAR.
- f) Report Request: The FIS-B report request enables an aircraft to request specific FIS data.

The FIS-B service is provided within a Transmission Volume (TV) which is defined as the area in which the FIS-B information applies.

Figure 4.3 illustrates the context for the airborne station supporting FIS-B.

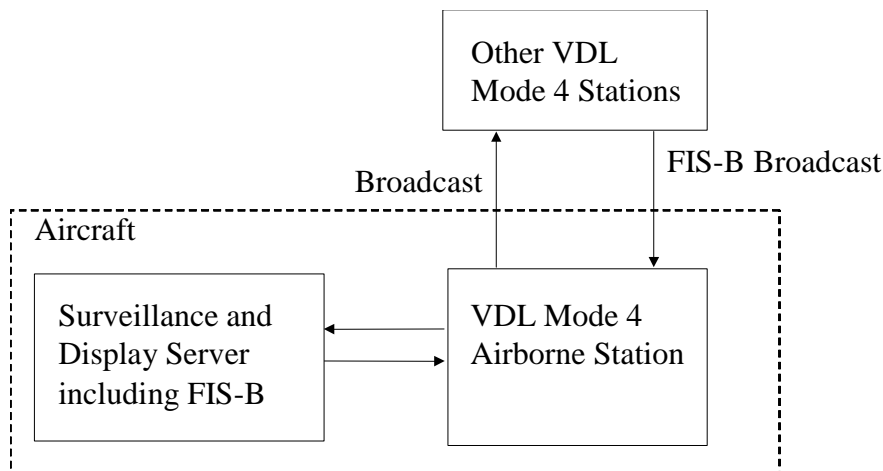


Figure 4.3: Ground station supporting FIS-B functions

A VDL Mode 4 airborne station supporting FIS-B receives broadcast FIS-B reports and passes on the data to a surveillance processor that includes FIS-B processing capability. The present document therefore includes requirements that apply to the part of the airborne equipment involved in FIS-B data processing, but this is not necessarily contained within the VDL Mode 4 airborne station.

The requirements for FIS-B are taken from the FIS-B Service Description produced by the Mediterranean UpDate and Near Update programmes (MEDUP and NUP).

4.5 GNSS Augmentation Service - Broadcast (GNS-B)

Ground stations may uplink GNSS augmentation data to aircraft in order to enhance the accuracy and integrity of GNSS. The GNSS Augmentation Service - Broadcast (GNS-B) is a VDL Mode 4 service intended to support a range of applications such as airport surface surveillance, terminal area and en-route operations, especially at high latitudes where space based augmentation systems are out of coverage. This service could be provided by a single VDL Mode 4 ground station or as an element within a ground network.

The present document presents minimum requirements for GNS-B that are derived from the GBAS specifications included in ICAO GNSS SARPs [7], GBAS MOPS [8], and from the GRAS Service Description developed by the MEDUP and NUP programmes, and include broadcast of:

- a) Pseudorange corrections (Message Type 1).
- b) GNS-B related data (Message Type 2).
- c) Final Approach Segment Data (Message Type 4).

Broadcast rates are dependent on the applications that should be supported and are not specified in the present document.

Figure 4.4 illustrates the context for the airborne station supporting GNS-B functions.

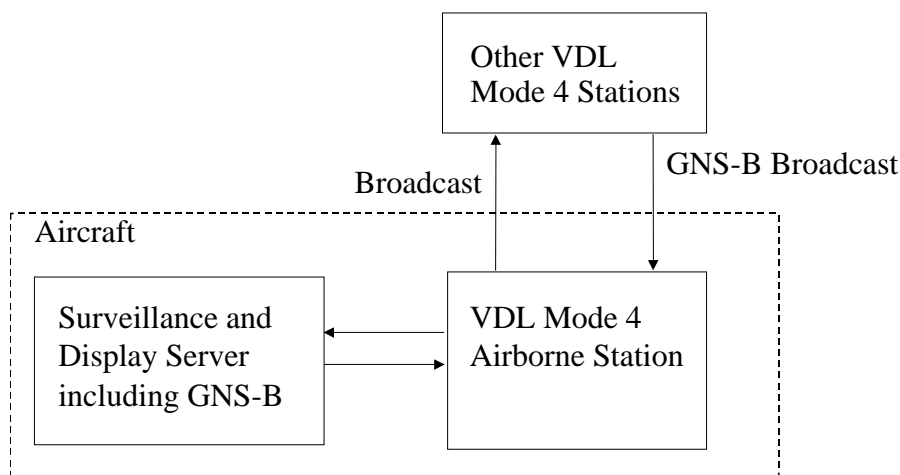


Figure 4.4: Airborne supporting GNS-B functions

A VDL Mode 4 airborne station supporting GNS-B receives broadcast GNS-B reports and passes on the data to a surveillance processor that includes GNS-B processing capability. The present document therefore includes requirements that apply to the part of the airborne equipment involved in GNS-B data processing, but this is not necessarily contained within the VDL Mode 4 airborne station.

5 Minimum performance specification under standard test conditions

5.1 Requirements for ADS-B

5.1.1 Information Field Formats

| Requirement reference | |
|-----------------------|--|
| 5.1.1.1 | The information field formats that have been assigned for use by the ADS-B application shall be as defined in table 5.1. |

Table 5.1: ADS-B information fields

| Information field ID (id) | ID extension 1 (id1) | ID extension 2 (id2) | Information field name |
|---------------------------|----------------------|----------------------|--|
| 0 hex | not present | not present | Basic |
| 1 hex | not present | not present | High dynamic |
| 2 hex | not present | not present | Full position |
| 3 hex | not present | not present | Basic ground |
| 4 hex | not present | not present | UTC time |
| 5 hex | not present | not present | Single Slot SVQ |
| 5 to 7 hex | not present | not present | Available for future use |
| 8 hex | not present | not present | Two slot TCP/SVQ |
| 9 hex | not present | not present | Single slot TCP |
| A hex | 0 hex | not present | Available for future use |
| A hex | 1 hex | not present | Aircraft data (call sign, category, status) |
| A hex | 2 to 9 hex | not present | Available for future use |
| A hex | A hex | 0 hex | High resolution |
| A hex | A hex | 1 to 9 hex | Available for future use |
| A hex | A hex | A hex | Extension (available for future use via further ID extension fields) |
| A hex | A hex | B to F hex | Available for future use |
| A hex | B to F hex | not present | Available for future use |
| B to E hex | not present | not present | Available for future use |
| F hex | not present | not present | No information field provided |

| Requirement reference | |
|-----------------------|---|
| 5.1.1.2 | For each information field, data shall be encoded as defined in table 5.2 to table 5.11. |
| NOTE: | Bits 1 and 2 in the last row of each of tables 5.2 to 5.11 are reserved for part of the reservation field (e.g. the periodic broadcast reservation field). If a reservation field is appended that does not use these bits, they should be set to zero. |

Table 5.2: Information field 0 hex - Basic

| Description | Octet | Bit number | | | | | | | |
|--|-------|-------------------|-------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| information field ID | 11 | x | x | x | x | 0 | 0 | 0 | 0 |
| Surveillance integrity level (sil) 6-bit latitude offset (lat6) | 12 | sil ₂ | sil ₁ | lat ₆ ₆ | lat ₆ ₅ | lat ₆ ₄ | lat ₆ ₃ | lat ₆ ₂ | lat ₆ ₁ |
| 6-bit longitude offset (lon6) baro rate/geo rate (br/gr) | 13 | res | br/gr | lon ₆ ₆ | lon ₆ ₅ | lon ₆ ₄ | lon ₆ ₃ | lon ₆ ₂ | lon ₆ ₁ |
| baro/geo offset (bgo) | 14 | altr ₉ | bgo ₇ | bgo ₆ | bgo ₅ | bgo ₄ | bgo ₃ | bgo ₂ | bgo ₁ |
| altitude rate (altr) | 15 | altr ₈ | altr ₇ | altr ₆ | altr ₅ | altr ₄ | altr ₃ | altr ₂ | altr ₁ |
| ground speed (gs) | 16 | gs ₈ | gs ₇ | gs ₆ | gs ₅ | gs ₄ | gs ₃ | gs ₂ | gs ₁ |
| ground track (gt) | 17 | gs ₁₁ | gs ₁₀ | gs ₉ | gt ₅ | gt ₄ | gt ₃ | gt ₂ | gt ₁ |
| | 18 | gt ₁₁ | gt ₁₀ | gt ₉ | gt ₈ | gt ₇ | gt ₆ | | |

NOTE: "x" denotes part of fixed data field.

Table 5.3: Information field 1 hex - High dynamic

| Description | Octet | Bit number | | | | | | | |
|--|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| information field ID | 11 | x | x | x | x | 0 | 0 | 0 | 1 |
| baro rate/geo rate (br/gr), baro/geo offset (bgo) | 12 | br/gr | bgo ₇ | bgo ₆ | bgo ₅ | bgo ₄ | bgo ₃ | bgo ₂ | bgo ₁ |
| altitude rate (altr) | 13 | altr ₈ | altr ₇ | altr ₆ | altr ₅ | altr ₄ | altr ₃ | altr ₂ | altr ₁ |
| Surveillance integrity level (sil) | 14 | altr ₉ | res | sil ₂ | sil ₁ | gs ₁₂ | gs ₁₁ | gs ₁₀ | gs ₉ |
| ground speed (gs) | 15 | gs ₈ | gs ₇ | gs ₆ | gs ₅ | gs ₄ | gs ₃ | gs ₂ | gs ₁ |
| 4-bit longitude offset (lon4), 4-bit latitude offset (lat4) | 16 | lon4 ₄ | lon4 ₃ | lon4 ₂ | lon4 ₁ | lat4 ₄ | lat4 ₃ | lat4 ₂ | lat4 ₁ |
| ground track (gt) | 17 | gt ₈ | gt ₇ | gt ₆ | gt ₅ | gt ₄ | gt ₃ | gt ₂ | gt ₁ |
| | 18 | gt ₁₂ | gt ₁₁ | gt ₁₀ | gt ₉ | res | res | | |

NOTE 1: "x" denotes part of fixed data field.
NOTE 2: "res" denotes currently unused.

Table 5.4: Information field 2 hex - Full position

| Description | Octet | Bit number | | | | | | | |
|------------------------------------|-------|-------------------|------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| information field ID | 11 | x | x | x | x | 0 | 0 | 1 | 0 |
| 6-bit latitude offset (lat6) | 12 | pid ₁₀ | pid ₉ | lat6 ₆ | lat6 ₅ | lat6 ₄ | lat6 ₃ | lat6 ₂ | lat6 ₁ |
| patch ID (pid) | 13 | pid ₈ | pid ₇ | pid ₆ | pid ₅ | pid ₄ | pid ₃ | pid ₂ | pid ₁ |
| baro/geo offset (bgo) | 14 | gt ₁₁ | bgo ₇ | bgo ₆ | bgo ₅ | bgo ₄ | bgo ₃ | bgo ₂ | bgo ₁ |
| 6-bit longitude offset (lon6) | 15 | gt ₁₀ | gt ₉ | lon6 ₆ | lon6 ₅ | lon6 ₄ | lon6 ₃ | lon6 ₂ | lon6 ₁ |
| ground track (gt) | 16 | gt ₈ | gt ₇ | gt ₆ | gt ₅ | gt ₄ | gt ₃ | gt ₂ | gt ₁ |
| ground speed (gs) | 17 | gs ₈ | gs ₇ | gs ₆ | gs ₅ | gs ₄ | gs ₃ | gs ₂ | gs ₁ |
| Surveillance Integrity Level (SIL) | 18 | gs ₁₁ | gs ₁₀ | gs ₉ | res | sil ₂ | sil ₁ | | |

NOTE: "x" denotes part of fixed data field.

Table 5.5: Information field 3 hex - Basic ground

| Description | Octet | Bit number | | | | | | | |
|--|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| information field ID | 11 | x | x | x | x | 0 | 0 | 1 | 1 |
| UTC hours (h) | 12 | res | res | res | h ₅ | h ₄ | h ₃ | h ₂ | h ₁ |
| UTC minute (min) | 13 | pid ₁₀ | pid ₉ | min ₆ | min ₅ | min ₄ | min ₃ | min ₂ | min ₁ |
| patch ID (pid) | 14 | pid ₈ | pid ₇ | pid ₆ | pid ₅ | pid ₄ | pid ₃ | pid ₂ | pid ₁ |
| baro/geo offset (bgo) | 15 | res | bgo ₇ | bgo ₆ | bgo ₅ | bgo ₄ | bgo ₃ | bgo ₂ | bgo ₁ |
| slot (slt) | 16 | slt ₈ | slt ₇ | slt ₆ | slt ₅ | slt ₄ | slt ₃ | slt ₂ | slt ₁ |
| 4-bit longitude offset (lon4), 4-bit latitude offset (lat4) | 17 | lon4 ₄ | lon4 ₃ | lon4 ₂ | lon4 ₁ | lat4 ₄ | lat4 ₃ | lat4 ₂ | lat4 ₁ |
| UTC second (sec) | 18 | sec ₆ | sec ₅ | sec ₄ | sec ₃ | sec ₂ | sec ₁ | | |

NOTE 1: "res" denotes currently unused.
NOTE 2: "x" denotes part of fixed data field.

Table 5.6: Information field 4 hex - UTC time

| Description | Octet | Bit number | | | | | | | |
|--|-------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| information field ID | 11 | x | x | x | x | 0 | 1 | 0 | 0 |
| UTC day (day) | 12 | res | res | res | day ₅ | day ₄ | day ₃ | day ₂ | day ₁ |
| UTC year (yr) | 13 | yr ₈ | yr ₇ | yr ₆ | yr ₅ | yr ₄ | yr ₃ | yr ₂ | yr ₁ |
| UTC hours (h), UTC month (mon) | 14 | h ₄ | h ₃ | h ₂ | h ₁ | mon ₄ | mon ₃ | mon ₂ | mon ₁ |
| UTC minute (min) | 15 | res | h ₅ | min ₆ | min ₅ | min ₄ | min ₃ | min ₂ | min ₁ |
| slot (slt) | 16 | slt ₈ | slt ₇ | slt ₆ | slt ₅ | slt ₄ | slt ₃ | slt ₂ | slt ₁ |
| 4-bit longitude offset (lon ₄), 4-bit latitude offset (lat ₄) | 17 | lon ₄ ₄ | lon ₄ ₃ | lon ₄ ₂ | lon ₄ ₁ | lat ₄ ₄ | lat ₄ ₃ | lat ₄ ₂ | lat ₄ ₁ |
| UTC second (sec) | 18 | sec ₆ | sec ₅ | sec ₄ | sec ₃ | sec ₂ | sec ₁ | | |

NOTE 1: "res" denotes currently unused.
NOTE 2: "x" denotes part of fixed data field.

Table 5.7: Information field 8 hex - Two slot TCP/SVQ

| Description | Octet | Bit number | | | | | | | |
|--------------------------------|-------|--------------------|--------------------|--------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| information field ID | 11 | x | x | x | x | 1 | 0 | 0 | 0 |
| TCP latitude (lat) | 12 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| | 13 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| base altitude (balt) | 14 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| longitude (lon) | 15 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| | 16 | ttg ₆ | ttg ₅ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| time to go (ttg) | 17 | ttg ₄ | ttg ₃ | ttg ₂ | ttg ₁ | res | res | res | res |
| TCP+1 latitude (lat) | 18 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| | 19 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| base altitude (balt) | 20 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| longitude (lon) | 21 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| | 22 | ttg ₆ | ttg ₅ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| time to go (ttg) | 23 | ttg ₄ | ttg ₃ | ttg ₂ | ttg ₁ | res | res | res | res |
| TCP+2 latitude (lat) | 24 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| | 25 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| base altitude (balt) | 26 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| longitude (lon) | 27 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| | 28 | ttg ₆ | ttg ₅ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| time to go (ttg) | 29 | ttg ₄ | ttg ₃ | ttg ₂ | ttg ₁ | res | res | res | res |
| TCP+3 latitude (lat) | 30 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| | 31 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| base altitude (balt) | 32 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| longitude (lon) | 33 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| | 34 | ttg ₆ | ttg ₅ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| time to go (ttg) | 35 | ttg ₄ | ttg ₃ | ttg ₂ | ttg ₁ | res | res | res | res |
| call sign left (csl) | 36 | csl ₈ | csl ₇ | csl ₆ | csl ₅ | csl ₄ | csl ₃ | csl ₂ | csl ₁ |
| | 37 | csl ₁₆ | csl ₁₅ | csl ₁₄ | csl ₁₃ | csl ₁₂ | csl ₁₁ | csl ₁₀ | csl ₉ |
| status (st) | 38 | st ₃ | st ₂ | st ₁ | csl ₂₁ | csl ₂₀ | csl ₁₉ | csl ₁₈ | csl ₁₇ |
| call sign right (csr) | 39 | csr ₈ | csr ₇ | csr ₆ | csr ₅ | csr ₄ | csr ₃ | csr ₂ | csr ₁ |
| | 40 | csr ₁₆ | csr ₁₅ | csr ₁₄ | csr ₁₃ | csr ₁₂ | csr ₁₁ | csr ₁₀ | csr ₉ |
| SVQ NAC, (nacv) | 41 | nacv ₃ | nacv ₂ | nacv ₁ | csr ₂₁ | csr ₂₀ | csr ₁₉ | csr ₁₈ | csr ₁₇ |

| Description | Octet | Bit number | | | | | | | |
|---|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| NAC _n (nacp), CDTI display capability (cdti), NIC _{baro} flag (nicb) ACAS operational flag (acas), Resolution Advisory active flag (ra) | 42 | nacp ₄ | nacp ₃ | nacp ₂ | nacp ₁ | cdti | nicb | acas | ra |
| current patch ID (pid) | 43 | pid ₈ | pid ₇ | pid ₆ | pid ₅ | pid ₄ | pid ₃ | pid ₂ | pid ₁ |
| 6-bit latitude offset (lat6) | 44 | pid ₁₀ | pid ₉ | lat6 ₆ | lat6 ₅ | lat6 ₄ | lat6 ₃ | lat6 ₂ | lat6 ₁ |
| 6-bit longitude offset (lon6), baro rate/geo rate (br/gr) | 45 | altr ₉ | br/gr | lon6 ₆ | lon6 ₅ | lon6 ₄ | lon6 ₃ | lon6 ₂ | lon6 ₁ |
| altitude rate (altr) | 46 | altr ₈ | altr ₇ | altr ₆ | altr ₅ | altr ₄ | altr ₃ | altr ₂ | altr ₁ |
| aircraft category (ac) SIL (sil) | 47 | res | sil ₂ | sil ₁ | ac ₅ | ac ₄ | ac ₃ | ac ₂ | ac ₁ |
| ground speed (gs) | 48 | gs ₈ | gs ₇ | gs ₆ | gs ₅ | gs ₄ | gs ₃ | gs ₂ | gs ₁ |
| ground track (gt) | 49 | gs ₁₁ | gs ₁₀ | gs ₉ | gt ₅ | gt ₄ | gt ₃ | gt ₂ | gt ₁ |
| | 50 | gt ₁₁ | gt ₁₀ | gt ₉ | gt ₈ | gt ₇ | gt ₆ | | |

NOTE 1: "x" denotes part of fixed data field.
NOTE 2: This burst definition allows a further octet of payload to complete the two slots.
NOTE 3: The detailed definition of the TCP information fields and their operational usage is still under discussion in the aviation community. As a consequence, the TCP format definition described above may be subject to future revision.

Table 5.8: Information field 9 hex - Single slot TCP

| Description | Octet | Bit number | | | | | | | |
|----------------------|-------|--------------------|--------------------|--------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| information field ID | 11 | x | x | x | x | 1 | 0 | 0 | 1 |
| base altitude (balt) | 12 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| TCP number (no) | 13 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | no ₂ | no ₁ | pid ₁₀ | pid ₉ |
| patch ID (pid) | 14 | pid ₈ | pid ₇ | pid ₆ | pid ₅ | pid ₄ | pid ₃ | pid ₂ | pid ₁ |
| latitude (lat) | 15 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| longitude (lon) | 16 | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ |
| | 17 | lon ₁₀ | lon ₉ | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ |
| time to go (ttg) | 18 | ttg ₆ | ttg ₅ | ttg ₄ | ttg ₃ | ttg ₂ | ttg ₁ | | |

NOTE: "x" denotes part of fixed data field.

Table 5.9: Information field 5 hex - Single Slot SVQ

| Description | Octet | Bit number | | | | | | | |
|--|-------|-------------------|-------------------|-------------------|------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| information field ID | 11 | x | x | x | x | 0 | 1 | 0 | 1 |
| NAC _p | 12 | 0 | 0 | 0 | 1 | nacp ₄ | nacp ₃ | nacp ₂ | nacp ₁ |
| NAC _v (nacv), SIL (sil), NIC _{baro} (nicb), ACAS operational flag (acas), Resolution Advisory active flag (ra) | 13 | nacv ₃ | nacv ₂ | nacv ₁ | sil ₂ | sil ₁ | nicb | acas | ra |
| status (st) and aircraft category (ac) | 14 | st ₃ | st ₂ | st ₁ | ac ₅ | ac ₄ | ac ₃ | ac ₂ | ac ₁ |
| Reserved for future definition | 15 | res | res | res | res | res | res | res | res |
| Reserved for future definition | 16 | res | res | res | res | res | res | res | res |
| Reserved for future definition | 17 | res | res | res | res | res | res | res | res |

NOTE: The detailed definition of the SVQ information fields and their operational usage is still under discussion in the aviation community. As a consequence, the SVQ format definition described above may be subject to future revision. Currently there is a discussion in the aviation community to provide further information such as describing the ADS-B position reference point, the length and width, the capability classes (e.g. information for TCAS/ACAS, CDTI, etc.) and the operational mode codes (receiving ATC services, etc.). There are spare octets to provide this information, when agreed in the future, but at this stage they are left unassigned to facilitate the future assignment.

Table 5.10: Information field A1 hex - Aircraft data

| Description | Octet | Bit number | | | | | | | |
|------------------------|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| information field ID | 11 | x | x | x | x | 1 | 0 | 1 | 0 |
| aircraft category (ac) | 12 | 0 | 0 | 0 | 1 | ac ₄ | ac ₃ | ac ₂ | ac ₁ |
| status (st) | 13 | ac ₅ | st ₃ | st ₂ | st ₁ | csl ₁₂ | csl ₁₁ | csl ₁₀ | csl ₉ |
| call sign left (csl) | 14 | csl ₈ | csl ₇ | csl ₆ | csl ₅ | csl ₄ | csl ₃ | csl ₂ | csl ₁ |
| | 15 | csl ₂₀ | csl ₁₉ | csl ₁₈ | csl ₁₇ | csl ₁₆ | csl ₁₅ | csl ₁₄ | csl ₁₃ |
| call sign right (csr) | 16 | csl ₂₁ | csr ₇ | csr ₆ | csr ₅ | csr ₄ | csr ₃ | csr ₂ | csr ₁ |
| | 17 | csr ₁₅ | csr ₁₄ | csr ₁₃ | csr ₁₂ | csr ₁₁ | csr ₁₀ | csr ₉ | csr ₈ |
| | 18 | csr ₂₁ | csr ₂₀ | csr ₁₉ | csr ₁₈ | csr ₁₇ | csr ₁₆ | | |

NOTE: "x" denotes part of fixed data field.

Table 5.11: Information field AA0 hex - High resolution

| Description | Octet | Bit number | | | | | | | |
|------------------------------------|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| information field ID | 11 | x | x | x | x | 1 | 0 | 1 | 0 |
| | 12 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| Surveillance Integrity Level (SIL) | 13 | res | res | sil ₂ | sil ₁ | gs ₁₂ | gs ₁₁ | gs ₁₀ | gs ₉ |
| ground speed (gs) | 14 | gs ₈ | gs ₇ | gs ₆ | gs ₅ | gs ₄ | gs ₃ | gs ₂ | gs ₁ |
| 8-bit longitude offset (lon8) | 15 | lon8 ₈ | lon8 ₇ | lon8 ₆ | lon8 ₅ | lon8 ₄ | lon8 ₃ | lon8 ₂ | lon8 ₁ |
| 8-bit latitude offset (lat8) | 16 | lat8 ₈ | lat8 ₇ | lat8 ₆ | lat8 ₅ | lat8 ₄ | lat8 ₃ | lat8 ₂ | lat8 ₁ |
| ground track (gt) | 17 | gt ₈ | gt ₇ | gt ₆ | gt ₅ | Gt ₄ | gt ₃ | gt ₂ | gt ₁ |
| turn indication (tind) | 18 | gt ₁₂ | gt ₁₁ | gt ₁₀ | gt ₉ | tind ₂ | tind ₁ | | |

NOTE: "x" denotes part of fixed data field.

| Requirement reference | |
|-----------------------|--|
| 5.1.1.3 | The values of the information field subfields shall be as defined in table 5.12. |

Table 5.12: Information field encoding (variable data field)

| Subfield | Range | Encoding | Notes |
|---|---------|--|-------|
| Surveillance Integrity Level (SIL) | 0 to 3 | Values 0, 1, 2, 3 in accordance with the four SIL categories specified in ADS-B MASPS RTCA/DO-242A [3] clause 2.1.2.15 | |
| Position Navigation Accuracy Category (NAC _p) | 0 to 11 | Values 0 through 11 in accordance with the twelve NAC _p categories specified in ADS-B MASPS RTCA/DO-242A [3] clause 2.1.2.13 | |
| Velocity Navigation Accuracy Category (NAC _v) | 0 to 4 | Values 0, 1, 2, 3, 4 in accordance with the twelve NAC _v categories specified in ADS-B MASPS RTCA/DO-242 A [3] clause 2.1.2.14 | |
| 4-bit latitude (lat4) | - | A high-resolution component to enhance the 12-bit low-resolution encoding transmitted in the fixed part (see EN 302 842-2 [5] clause 5.4.2.3). Encoding of this high-resolution component is described in clause 5.1.6 | |
| 6-bit latitude (lat6) | - | See encoding for lat4 | |
| 8-bit latitude (lat8) | - | See encoding for lat4 | |
| 4-bit longitude (lon4) | - | A high-resolution component to enhance the 14-bit low-resolution encoding transmitted in the fixed part (see EN 302 842-2 [5] clause 5.4.2.3). Encoding of this high-resolution component is described in clause 5.1.6 | |
| 6-bit longitude (lon6) | - | See encoding for lon4 | |

| Subfield | Range | Encoding | Notes |
|---|----------------------------------|---|---|
| 8-bit longitude (lon8) | - | See encoding for lon4 | |
| baro/geo offset (bgo) | 0 to 127 | bgo = barometric - geometric altitude Encoding as in table 5.13 | |
| baro rate/geo rate (br/gr) | Binary | 0 = altitude rate is barometric altitude rate 1 = altitude rate is geometric altitude rate | |
| altitude rate (altr) | -32 100 fpm to +32 100 fpm | Bit $altr_9$ encodes the sign of altitude rate of change with 0 = climb and 1 = descend. $altr_9$ shall be set to 0 if the magnitude of altitude rate is unknown Bits $altr_8 \dots altr_1$ encode the magnitude of altitude rate of change as specified in table 5.15. | |
| ground speed (gs) | 0 to \geq 11 256 knots | Encoding as in table 5.14 | Range is 0 to 3 069 knots for 11 bits and 0 to 11 256 for 12 bits. Resolution steps from 1 knot to 4 knots. Note that bit 12 is only available in the high dynamic variable part. |
| ground track (gt) | 0° to 359,912° | due North $\pm \frac{1}{2}N+1$ degrees coded as 0 and decoded as due North Resolution is $360/2^N$ degrees | Ground track is the same as true track. N is the number of bits (either 11 or 12) assigned in the variable field for ground track. |
| turn indication (tind) | 0 to 3 | 0 = Unknown, 1 = Left, 2 = Right and 3 = Straight | The threshold between straight and turning is an operational issue that will be specified elsewhere. |
| patch ID (pid) | - | Encoding is described in clause 5.1.6.4 | |
| UTC year (yr) | 1 to 255 | current year - 1970, 0 = N/A | |
| UTC month (mon) | 1 to 12 | integer months | |
| UTC day (day) | 1 to 31 | integer days, 00 = N/A | |
| UTC hours (h) | 0 to 23 | integer hours | |
| UTC minute (min) | 0 to 59 | integer minutes | |
| UTC second (sec) | 0 to 60 | integer seconds | Seconds run up to 60 to allow for leap seconds. |
| slot (slt) | 0 to 255 | integer slots, 0 indicates the first slot in the second frame | |
| TCP number (no) | 0 to 3 | 0 = current 1 = next 2 = next + 1 3 = next + 2 | |
| TCP time to go (ttg) | 0 to 63 | Indicates the time to reach the indicated TCP from either the current position (no - 0) or from the previous TCP (no = 1, 2 or 3). Encoding as in table 5.18 | |
| call sign left (csl), call sign right (csr) | | Encoding for call sign: 1) Call sign shall be left justified 2) Only valid characters are A-Z, 0 - 9 and null: Assign A- Z = 0 - 25, 0 - 9 = 26 - 35, null = 36 3) Call sign shall be an eight character string " $c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8$ " 4) $csl = c_1 37^3 + c_2 37^2 + c_3 37 + c_4$ 5) $csr = c_5 37^3 + c_6 37^2 + c_7 37 + c_8$ | |
| aircraft category (ac) | 0 to 31 | Encoding as in table 5.16 | |
| status (st) | 0 to 7 | Encoding as in table 5.17 | |
| CDTI flag (cdti) | Binary | 0 = not CDTI equipped. 1 = CDTI equipment installed and operational. | |

| Requirement reference | |
|-----------------------|--|
| 5.1.1.4 | For TCP transmission, position encoding shall use the Patch ID encoding described in clause 5.1.6.4. |
| 5.1.1.5 | For transmission of the single slot TCP variable part (information field 9 hex), bits 1 and 2 of the longitude subfield shall not be sent. |
| 5.1.1.6 | On receipt of a single slot TCP variable part, the value of bits 1 and 2 shall be assumed to be equal to 0. |
| 5.1.1.7 | The encoding of the baro/geo offset subfield shall be as defined in table 5.13. |

Table 5.13: Baro/geo offset encoding (variable data field)

| Barometric/geometric altitude offset of transmitting station (ft) | | Transmitted value of bgo | Decoded offset (ft) | |
|---|----------------------------|--------------------------|--------------------------|---------------------|
| base alt \leq 24 012.5 | 24 012.5 < base alt | | base alt \leq 24 012.5 | 24 012.5 < base alt |
| bgo < -2 075 | bgo < -5 050 | 1 | < -2 075 | < -5 050 |
| -2 075 \leq bgo < -2 025 | -5 050 \leq bgo < -4 950 | 2 | -2 050 | -5 000 |
| -2 025 \leq bgo < -1 975 | -4 950 \leq bgo < -4 850 | 3 | -2 000 | -4 900 |
| -1 975 \leq bgo < -1 925 | -4 850 \leq bgo < -4 750 | 4 | -1 950 | -4 800 |
| -1 925 \leq bgo < -1 875 | -4 750 \leq bgo < -4 650 | 5 | -1 900 | -4 700 |
| -1 875 \leq bgo < -1 825 | -4 650 \leq bgo < -4 550 | 6 | -1 850 | -4 600 |
| ↓ | ↓ | ↓ | ↓ | ↓ |
| 3 975 \leq bgo < 4 025 | 7 050 \leq bgo < 7 150 | 123 | 4 000 | 7 100 |
| 4 025 \leq bgo < 4 075 | 7 150 \leq bgo < 7 250 | 124 | 4 050 | 7 200 |
| 4 075 \leq bgo < 4 125 | 7 250 \leq bgo < 7 350 | 125 | 4 100 | 7 300 |
| bgo \geq 4 125 | bgo \geq 7 350 | 126 | > 4 125 | > 7 350 |
| reserved | | 127 | error | |
| offset unknown | | 0 | offset unknown | |

| Requirement reference | |
|-----------------------|--|
| 5.1.1.8 | The encoding of the ground speed subfield shall be as defined in table 5.14. |

Table 5.14: Ground speed encoding (variable data field)

| Actual speed over ground of transmitting station (knots) | Transmitted value of gs (decimal equivalent of field) | Decoded speed over ground (knots) |
|--|---|-----------------------------------|
| unknown | 0 | unknown |
| $0 \leq \text{speed} < 0.5$ | 1 | 0 |
| $0.5 \leq \text{speed} < 1.5$ | 2 | 1 |
| $1.5 \leq \text{speed} < 2.5$ | 3 | 2 |
| $2.5 \leq \text{speed} < 3.5$ | 4 | 3 |
| $3.5 \leq \text{speed} < 4.5$ | 5 | 4 |
| ↓ | ↓ | ↓ |
| $1\ 021.5 \leq \text{speed} < 1\ 022.5$ | 1 023 | 1 022 |
| $1\ 022.5 \leq \text{speed} < 1\ 024$ | 1 024 | 1 023 |
| $1\ 024 \leq \text{speed} < 1\ 026$ | 1 025 | 1 025 |
| $1\ 026 \leq \text{speed} < 1\ 028$ | 1 026 | 1 027 |
| ↓ | ↓ | ↓ |
| $3\ 068 \leq \text{speed} < 3\ 070$ | 2 047 | 3 069 |
| $3\ 070 \leq \text{speed} < 3\ 074$ | 2 048 | 3 072 |
| $3\ 074 \leq \text{speed} < 3\ 078$ | 2 049 | 3 076 |
| $3\ 078 \leq \text{speed} < 3\ 082$ | 2 050 | 3 080 |
| ↓ | ↓ | ↓ |
| $11\ 250 \leq \text{speed} < 11\ 254$ | 4 093 | 11 252 |
| $11\ 254 \leq \text{speed} < 11\ 258$ | 4 094 | 11 256 |
| $11\ 258 \leq \text{speed}$ | 4 095 | more than or equal to 11 258 |

| Requirement reference | |
|-----------------------|---|
| 5.1.1.9 | The encoding of the magnitude part of the altitude rate subfield (bits 1 to 8) shall be as defined in table 5.15. |

Table 5.15: Altitude rate magnitude encoding and decoding

| Actual altitude rate of transmitting station (fpm) | Transmitted decimal value of $\text{altr}_8 \dots \text{altr}_1$ | Decoded magnitude of altitude rate (fpm) |
|--|--|--|
| unknown | 0 | unknown |
| $ \text{arate} < 50$ | 1 | 0 |
| $50 \leq \text{arate} < 150$ | 2 | 100 |
| $150 \leq \text{arate} < 250$ | 3 | 200 |
| $250 \leq \text{arate} < 350$ | 4 | 300 |
| ↓ | ↓ | ↓ |
| $19\ 950 \leq \text{arate} < 20\ 050$ | 201 | 20 000 |
| $20\ 050 \leq \text{arate} < 20\ 300$ | 202 | 20 200 |
| $20\ 300 \leq \text{arate} < 20\ 500$ | 203 | 20 400 |
| ↓ | ↓ | ↓ |
| $29\ 900 \leq \text{arate} < 30\ 100$ | 251 | 30 000 |
| $30\ 100 \leq \text{arate} < 30\ 500$ | 252 | 30 250 |
| $30\ 500 \leq \text{arate} < 31\ 000$ | 253 | 30 750 |
| $31\ 000 \leq \text{arate} < 32\ 000$ | 254 | 31 500 |
| $32\ 000 \leq \text{arate} $ | 255 | more than or equal to 32 000 |

| | |
|------------------------------|---|
| Requirement reference | |
| 5.1.1.10 | The encoding of the aircraft category subfield shall be as defined in table 5.16. |

Table 5.16: Aircraft category encoding

| Encoded value | Aircraft/vehicle category | Comments |
|----------------------------|---------------------------------------|---|
| 0 | light a/c | 7 000 kgs (15 500 lbs) or less |
| 1 | reserved | |
| 2 | medium a/c | more than 7 000 kgs (15 500 lbs) and less than 136 000 kg (300 000 lbs) |
| 3 | reserved | |
| 4 | heavy a/c | 136 000 kgs (300 000 lbs) or more |
| 5 | highly manoeuvrable and high speed | > 5 g acceleration capability and > 400 knots |
| 6, 7, 8 | reserved | |
| 9 | rotorcraft | |
| 10 | glider/sailplane | |
| 11 | lighter than air | |
| 12 | unmanned aerial vehicle | |
| 13 | space/transatmospheric vehicle | |
| 14 | ultra -light/hang-glider/para -glider | |
| 15 | parachutist/skydiver | |
| 16, 17, 18 | reserved | |
| 19 | surface vehicle - emergency vehicle | |
| 20 | surface vehicle - service vehicle | |
| 21 | fixed ground or tethered obstruction | |
| 22, 23 | reserved | |
| 24, 25, 26, 27, 28, 29, 30 | reserved | |
| 31 | unknown | |

| | |
|------------------------------|---|
| Requirement reference | |
| 5.1.1.11 | The encoding of the aircraft status subfield shall be as defined in table 5.17. |

Table 5.17: Aircraft status (emergency/priority status) encoding

| Encoded value | Status |
|---------------|---------------------------|
| 0 | no emergency/not reported |
| 1 | general emergency |
| 2 | lifeguard/medical |
| 3 | minimal fuel |
| 4 | no communications |
| 5 | unlawful interference |
| 6, 7 | reserved |

| | |
|------------------------------|--|
| Requirement reference | |
| 5.1.1.12 | The encoding of the time to go (ttg) subfield shall be as defined in table 5.18. |

Table 5.18: Time to go subfield encoding

| Actual time to go (minutes) | Transmitted decimal value of ttg | Decoded magnitude of time to go (minutes) |
|------------------------------|----------------------------------|---|
| unknown | 0 | No time to go information available |
| time to go < 0,125 | 1 | 0 |
| 0,125 ≤ time to go < 0,375 | 2 | 0,25 |
| 0,375 ≤ time to go < 0,625 | 3 | 0,5 |
| ↓ | ↓ | ↓ |
| 15,125 ≤ time to go < 15,375 | 62 | 15,25 |
| 15,375 ≤ time to go | 63 | more than or equal to 15,375 |

5.1.2 ADS-B Request

5.1.2.1 ADS-B request format

| Requirement reference | |
|-----------------------|--|
| 5.1.2.1.1 | To request that a station transmit an ADS-B report consisting of a synchronization burst a station shall transmit a general request burst with $r-mi_1 = 0$. |
| 5.1.2.1.2 | To request that a station transmit an ADS-B report consisting of a synchronization burst a station shall include the auxiliary information as shown in tables 5.19 to 5.21, as appropriate given the values of the sleep and auto bits in octet 5. |
| 5.1.2.1.3 | The information subfields shall be encoded according to table 5.22. |
| 5.1.2.1.4 | Parameter blocks shall be included only if the respective parameter flag bit is 1. |
| 5.1.2.1.5 | Parameter blocks shall be included in the order (most significant to least significant) of the parameter flag bits. |

5.1.2.2 Sleep Mode

| Requirement reference | |
|-----------------------|--|
| 5.1.2.2.1 | If the requesting station desires the responding station to respond at one specified rate as a default, but a different specified rate in the event certain position or velocity deviation thresholds are exceeded, the requesting station shall set the sleep bit equal to 1 and octets 6 to 7 in table 5.20 shall be sent. |
| 5.1.2.2.2 | Otherwise, the sleep bit shall be set = 0 and octets 6 to 7 in table 5.20 shall not be sent. |

5.1.2.3 Automatic Selection of Variable Information Fields

| Requirement reference | |
|-----------------------|---|
| 5.1.2.3.1 | If the requesting station desires the responding station to respond with synchronization burst variable parts selected autonomously by the responding station, the requesting station shall set the auto bit = 0. |
| 5.1.2.3.2 | and octet k in table 5.21 shall not be sent. |
| 5.1.2.3.3 | Otherwise, the station shall set the auto bit = 1. |
| 5.1.2.3.4 | and the desired variable part ID shall be encoded as indicated in table 5.21. |

Table 5.19: ADS-B request bit encoding

| Description | Octet | Bit number | | | | | | | |
|---|--------|--------------------------|------|--------------------|--------------------|---|---|---|---|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| burst ID autonomous monitoring (sleep), autonomous information (auto), requested base altitude (r-b/a) | 5 | sleep | auto | r-b/a ₂ | r-b/a ₁ | 0 | 0 | 0 | 1 |
| | 6 to m | see tables 5.20 and 5.21 | | | | | | | |

Table 5.20: ADS-B request bit encoding for sleep mode parameters when sleep bit = 1

| Description | Octet | Bit number | | | | | | | |
|---|-------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| secondary reporting rate (snr), maximum sleep velocity (vel) | 6 | snr ₄ | snr ₃ | snr ₂ | snr ₁ | vel ₄ | vel ₃ | vel ₂ | vel ₁ |
| maximum sleep position (pos) | 7 | pos ₈ | pos ₇ | pos ₆ | pos ₅ | pos ₄ | pos ₃ | pos ₂ | pos ₁ |

Table 5.21: ADS-B request bit encoding for auto parameters when auto bit = 1

| Description | Octet | Bit number | | | | | | | |
|---|-------|-----------------|-----------------|------------------|------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| length (lg), auxiliary data (aux), requested information field ID (r-id) (see note) | k | lg ₂ | lg ₁ | aux ₂ | aux ₁ | r-id ₄ | r-id ₃ | r-id ₂ | r-id ₁ |

NOTE: The r-id field may continue into additional octets depending on the variable part requested.

Table 5.22: ADS-B request field encoding

| Subfield | Range | Encoding | Out of Scale | Notes |
|---------------------------------|-------------------------|---|------------------------------------|---|
| sleep | Boolean | 0 = no sleep mode; 1 = sleep mode | n/a | if 1, additional information describing sleep mode parameters is provided as a first additional information subfield starting in octet 6. |
| autonomous information (auto) | Boolean | 0 = autonomously select transmitted information field 1 = provide requested information field only | n/a | if 1, additional information describing requested information is provided as an additional information subfield starting in octet 6 (if sleep = 0 implying lack of sleep mode parameters), or octet 8 (if sleep = 1). |
| requested information ID (r-id) | Binary; variable length | See table 5.2 | A value not contained in table 5.2 | Requests the information field identity contained in the variable data field. |
| maximum sleep velocity (vel) | 1 to 15 knots | integer knots | 0= ignore | |
| maximum sleep position (pos) | 1 to 255 metres | integer metres | 0= ignore | |
| secondary reporting rate (snr) | same as nr | see table 5.28 in EN 302 842-2 [5] | n/a | |
| parameter group length (lg) | 0 to 3 | Binary | n/a | Indicates number of additional octets in parameter group (e.g. a parameter group comprising a single octet is encoded as 0). |
| auxiliary parameters (aux) | 0 to 3 | If r-id = 0 or 1 hex, then: 00 = baro rate; 01 = geo rate 10 = reserved 11 = do not care. If r-id = 9 _{hex} , then: 00 = current TCP; 01 = next TCP, 10 = next + 1 TCP, 11 = next + 2 TCP. If r-id = 2, 3, 4, 8, A1, or AA0 hex then: aux = 00 on transmit and ignored on receipt | | Interpretation depends on value of r-id field. Unassigned values are reserved. |
| requested base altitude (r-b/a) | 0 to 3 | See EN 302 842-2 [5] table 5.76 | | |

5.1.3 Default ADS-B Reporting

| Requirement reference | |
|--------------------------------------|--|
| 5.1.3.1 | In the absence of an ADS-B request from a peer station or a request from the ADS-B application in the station, the station shall transmit at least 12 synchronization bursts per minute averaged over the two GSCs. |
| Default transmission sequence | |
| 5.1.3.1a | In default mode, and in the absence of any other instructions (e.g. a channel management command), and if the aircraft has TCPs to transmit, and if the aircraft is at an altitude exceeding 25ft above ground level, an ADS-B transceiver shall transmit a series of sync bursts consisting of 11 basic sync bursts followed by one 2-slot TCP message. |
| 5.1.3.1b | In default mode, and in the absence of any other instructions (e.g. a channel management command), and if the aircraft has no TCPs to transmit, an ADS-B transceiver shall transmit basic sync bursts. |
| Recommendation | |
| 5.1.3.2 | A station should aim to distribute its transmissions evenly between the two GSCs. |

5.1.4 ADS-B Procedures

| Requirement reference | |
|---|---|
| ADS-B request procedures | |
| 5.1.4.1 | A station issuing an ADS-B request shall set the auto bit to 1 and include the r-id field when requesting a specific information field. |
| 5.1.4.2 | The requesting station shall set the auto bit to 0 and not include the r-id field when it desires that the responding unit determine which information field is the most important at any point in time. |
| 5.1.4.3 | The requesting station shall set the sleep bit to 0 when requesting a single broadcast response (e.g. via a unicast request reservation). |
| 5.1.4.4 | The requesting station shall set the sleep bit to 1 and include position and velocity thresholds when it wants the responding station to transmit a synchronization burst in directed slots at one rate, but transmit at a higher rate under certain circumstances. |
| 5.1.4.4a | A mobile station in receipt of an ADS-B request in which the sleep bit is set to 1, shall monitor its position and velocity to determine if the station has exceeded either of two thresholds: (a) moving more than pos metres from the position reported in the last directed report; or (b) moving more than vel knots. |
| 5.1.4.4b | If pos is zero, then the position test shall be ignored. |
| 5.1.4.4c | If vel is zero, then the velocity test shall be ignored. |
| 5.1.4.4d | A station which exceeded either the position or velocity threshold, shall begin to transmit autonomously using the incremental broadcast procedures until one of the following occurs: a) it receives a directed request reservation for the frequency on which it is transmitting autonomously with the or bit set to 1 (both the autonomous incremental transmissions and the directed periodic transmissions shall be affected); b) it is transmitting autonomous synchronization bursts because it exceeded the position threshold, but not the velocity threshold, and it subsequently transmitted a directed synchronization burst (in this case, the station shall not make an incremental reservation past the slot in which it will transmit the directed synchronization burst); c) it is transmitting autonomous synchronization bursts because it exceeded the velocity threshold, and it subsequently transmitted a directed synchronization burst after its velocity had subsided below the threshold. |
| Transmission of time synchronization request | |
| 5.1.4.5 | A station requesting the UTC time synchronization information field (information field ID 24) shall only transmit a request to a station that has announced that it is operating with a primary time source (tfom = 0 or 1). |

| Requirement reference | |
|-----------------------|--|
| | Information field priority |
| 5.1.4.6 | When a station is requested to transmit a series of sync bursts with different variable part content, information fields 0, 1, 2, 3, 4 and A1 hex shall have priority over other information fields. |
| | TCP/SVQ change procedures |
| 5.1.4.6a | When the status of its first TCP changes, a mobile station shall autonomously transmit a series of synchronization bursts containing the single-slot TCP variable part (see table 5.8). |
| 5.1.4.6b | The update interval for these transmissions shall be tci seconds |
| 5.1.4.6c | and this high reporting rate shall be maintained for a period of tcd seconds, as defined in table 5.22a. |

Table 5.22a: TCP update rate parameters

| Symbol | Parameter name | Minimum | Maximum | Default |
|--------|-------------------------------|---------|---------|---------|
| tci | TCP high update interval | 1 s | 60 s | 2 s |
| tcd | TCP high update rate duration | 1 s | 60 s | 12 s |

| Requirement reference | |
|-----------------------|--|
| 5.1.4.6d | When its SVQ status changes, a mobile station shall autonomously transmit a series of synchronization bursts containing the single-slot SVQ variable part (see table 5.9). |
| 5.1.4.6e | The update interval for these transmissions shall be svi seconds and this high reporting rate shall be maintained for a period of svd seconds, as defined in table 5.22b. |

Table 5.22b: SVQ update rate parameters

| Symbol | Parameter name | Minimum | Maximum | Default |
|--------|-------------------------------|---------|---------|---------|
| svi | SVQ high update interval | 1 s | 60 s | 2 s |
| svd | SVQ high update rate duration | 1 s | 60 s | 12 s |

| Requirement reference | |
|-----------------------|--|
| 5.1.4.6f | A mobile station which is transmitting directed synchronization bursts which, as part of a regular periodic series of synchronization bursts, include the two slot TCP/SVQ variable part defined in table 5.7, shall indicate when there is a change to any of its TCPs or to its State Vector Quality (SVQ) indicators by setting the TCP/SVQ change flag (see clause 5.4.2.3 in EN 302 842-2 [5]) to zero in all directed synchronization bursts transmitted by the station. |
| 5.1.4.7 | When a mobile has set its TCP/SVQ change flag to 0, only ground stations shall be allowed to respond by issuing an ADS-B request burst requesting that the mobile transmit a synchronization burst containing the two slot TCP/SVQ variable part. |
| 5.1.4.8 | The mobile station shall set the TCP/SVQ change flag to 1 when it has transmitted updated TCP and/or SVQ information using the two slot TCP/SVQ variable part. |
| 5.1.4.9 | A station which is transmitting directed synchronization bursts which do not include the two slot TCP/SVQ variable part in the regular periodic series of synchronization bursts, shall set the TCP/SVQ change flag to 1. |

5.1.5 CTRL Parameters

5.1.5.1 Directory of Service (DOS) message

| Requirement reference | |
|-----------------------|--|
| 5.1.5.1.1 | Directory of service information shall be transmitted by a ground station only using the CTRL DOS parameter defined in EN 302 842-2 [5] clauses 5.4.3.4.2 to 5.4.3.4.12. |
| 5.1.5.1.2 | The application fields shall be encoded as defined in table 5.23. |

Table 5.23: Encoding of application fields

| Encoding (decimal equivalent) | Implied meaning |
|-------------------------------|---------------------------------|
| 0 | ADS-B |
| 1 | TIS-B |
| 2 | Secondary navigation capability |
| 3 | FIS-B |
| 4 | GNS-B |

| Requirement reference | |
|-----------------------|--|
| 5.1.5.1.3 | The service information (si) field for service information type (sit) field equal to hex 00 shall be as defined in table 5.24. |

Table 5.24: Encoding of service information field for sit = hex 00

| Service information (si) field bit number | Service | Equivalent application field (see table 5.23) |
|---|---------------------------------|---|
| 1 | ADS-B | 0 |
| 2 | TIS-B | 1 |
| 3 | Secondary navigation capability | 2 |
| 4 | FIS-B | 3 |
| 5 | GNS-B | 4 |
| 6 to 8 | reserved | |

5.1.5.2 Channel Management Parameter

| Requirement reference | |
|-----------------------|---|
| 5.1.5.2.1 | The channel management parameter shall have a parameter ID equal to C6h. |
| 5.1.5.2.2 | If the channel management parameter is included within a UCTRL DLPDU with the ucid subfield set to 2, the channel management parameter ID and parameter length shall be omitted and no other parameter included in the UCTRL_DLPDU. |
| 5.1.5.2.3 | The channel management parameter shall be divided into component blocks as described in table 5.25. |
| 5.1.5.2.4 | If present, each component block shall appear in the channel management parameter in the order shown in table 5.25. |

Table 5.25: Component blocks of channel management parameter

| Blocks | Description |
|-------------------------------|---|
| Header block | Contains the CTRL header and the non-repetitive information for each of the sub-elements. |
| Destination block | Contains a list of zero or more destinations or a geographic region, along with the timeout and slot offset from the current slot to the start of the script (zero destinations included implies broadcast). |
| Transmission definition block | Contains from 0 to 15 transmission definitions in an unordered list. Each definition describes a particular desired transmission (one not already specified in SARPs) and all relevant options. The location of a particular transmission block in this list defines an implicit pointer used in the script block (i.e. thereby specifying a particular message). |
| Frequency block | Contains from 1 to 7 frequencies in an ordered list and optional sleep parameters to use on each frequency. The location of a particular frequency block in this list defines an implicit pointer used in the script block (i.e. thereby specifying a particular frequency). |
| Parameter block | Contains QoS and other transmission parameters to use with the elements of this script. |
| Script block | Contains a repeat rate and from 1 to N script elements in an ordered list. Each script element consists of an optional repeat count, a frequency block pointer, and a transmission block pointer. |

| Requirement reference | Header block encoding |
|------------------------------|--|
| 5.1.5.2.5 | The channel management parameter header block shall appear first in the channel management parameter and be encoded as defined in table 5.26 with subfield encodings as defined in table 5.27. |

Table 5.26: Header block bit encoding

| Description | Octet | Bit number | | | | | | | |
|--|--------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|------------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| directed timeout (dt) | m+2 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| directed offset (do) | m+3 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| destination count (dc) | m+4 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| command set ID (csid), transmission definition count (tc) | m+5 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| frequency count (fc), script duration type (styp), script rate (sr) | m+6 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| transmit parameter count (pc) | m+7 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| script count (sc) | m+8 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |

Table 5.27: Header block field encoding

| Subfield | Encoding | Notes |
|--|---|---|
| directed timeout (dt) | See EN 302 842-2 [5], table 5.30 | |
| directed offset (do) | See EN 302 842-2 [5], table 5.30 | |
| destination count (dc) | 0: destination block absent (see clause 5.1.5.2.9) 1 to 255: number of octets in destination block | 0 implies broadcast |
| command set ID (csid) | 0 to 14: valid csid identifiers 15: cancel all prior commands | identifier specifying this command (see note) |
| transmission definition count (tc) | 0: transmission block absent 1 to 15: number of elements in the transmission definition block | |
| frequency count (fc) | 0: no freq listed (csid = 15) 1 to 7: number of frequencies | The number of elements in the frequency block |
| script duration type (styp) | styp = 0: interpret sr as the number of times the entire script repeats per minute styp = 1: interpret sr as the number of minutes until script repeats | |
| script rate (sr) | 0 to 15 Interpreted based on styp For styp = 0, sr is encoded per nr in EN 302 842-2 [5] table 5.28 For styp = 1, sr is encoded as minutes between 2 (encoded as 0) and 17 minutes | |
| script count (sc) | 1 (encoded as 0) to 256 | Number of bytes in script block |
| transmit parameter count (pc) | 0 to 63 | Number of bytes in the transmit parameter block |
| <p>NOTE: The command set ID allows a ground station to uplink several (up to 15) separate commands to each aircraft or set of aircraft (including all aircraft in view) which are each processed separately. A transmission with csid = x overrides operations specified in a previous transmission with csid = x, but does not affect operations dictated by previous transmissions with csid not equal to x. Hence a ground station can command multiple sets of messages (each set containing various types of transmissions), and the transmissions can have different report periods from one set to another. An example is a sequence of sync bursts on two or more frequencies defined by command set csid = 0, and a separate sequence of TCPs on the same or different frequency(ies) defined by command set csid = 1. The use of different command sets allows directed-slot operation for both command sets even though the two sets of commanded transmissions may have different burst lengths (e.g., one slot for sync bursts and two slots for TCPs).</p> | | |

| Requirement reference | |
|-----------------------------------|---|
| 5.1.5.2.6 | The number of elements in the script (total number of transmissions over one repeat cycle) shall be designated NES. |
| 5.1.5.2.7 | The number of transmissions per minute shall be designated NTM. |
| Destination block encoding | |
| 5.1.5.2.8 | The destination block shall consist of zero destinations (a broadcast), one or more addressed destinations, or a regional definition, as defined in the following clauses. |
| Broadcast destination | |
| 5.1.5.2.9 | If the destination count (dc) in the header block is zero, then the channel management CTRL parameter shall apply to all stations. |
| Addressed destination | |
| 5.1.5.2.10 | If bit 8 of the first octet in the destination block is a zero, then the destination block shall consist of one or more addresses per table 5.28 with encoding defined in table 5.29. |
| 5.1.5.2.10a | If none of the addresses matches the receiver, then further processing of this parameter shall terminate. |

Table 5.28: Addressed destination block bit encoding

| Description | Octet | Bit number | | | | | | | |
|-------------------------|-------|-----------------|------------------|------------------|------------------|------------------|-----------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| Individual offset (ido) | 1 | 0 | ido ₄ | ido ₃ | ido ₂ | ido ₁ | d ₂₇ | d ₂₆ | d ₂₅ |
| destination address (d) | 2 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| | 3 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| | 4 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |

Table 5.29: Addressed destination block field encoding

| Subfield | Encoding | Notes |
|-------------------------|---|---------------------------------|
| Individual offset (ido) | 0 to 15 for the first address in the addressed destination block 0 to 31 otherwise | See description of usage below. |
| destination address (d) | See EN 302 842-2 [5] clause 5.3.1.2 | |

| Requirement reference | |
|---------------------------------|---|
| Directed Rate Operations | |
| 5.1.5.2.11 | A mobile station receiving a channel management parameter with the individual offset (ido) set equal to zero in a destination block with the station's address, shall operate in directed rate mode. |
| 5.1.5.2.12 | A mobile station in receipt of a channel management parameter with destination count (dc = 0) in the header block shall operate in directed rate mode. |
| Directed Slot Operations | |
| 5.1.5.2.13 | A mobile station receiving a channel management parameter with the individual offset (ido) not set equal to zero in a destination block with the station's address shall operate in directed slot mode. |
| 5.1.5.2.14 | The sum of the individual offset parameter ido for each addressed station and directed offset (do) subfield shall indicate the number of slots by which the transmissions of that station are to be offset from the first slot of the CTRL command. |
| 5.1.5.2.15 | Void. |
| Regional multicast | |
| 5.1.5.2.16 | If bit 8 of the first octet of the destination block is a one, then the destination block shall instead consist of a regional multicast per table 5.30 with encodings defined in table 5.31. |
| 5.1.5.2.16a | Stations operating in directed-slot mode from a previous channel management parameter, or from a directed request (see EN 302 842-2 [5], clause 5.2.16), shall not process regional multicast commands. |
| 5.1.5.2.17 | A station shall determine that it is in the defined region if Condition 1 is true and either Condition 2a or Condition 2b is true: Condition 1: altitude compliance. - the station altitude (using baro altitude if available, otherwise geo altitude) is greater than or equal to the lower altitude and less than or equal to the upper altitude; Condition 2a: radial range compliance when there is at least one vertex k with radial k less than or equal to the station's radial from the centre of the defined region. - the station distance from the centre of the region, assuming a spherical Earth, is less than or equal to the distance k associated with the vertex k having greatest radial k less than or equal to the station's radial from the centre of the defined region, or Condition 2b: radial range compliance when there is no vertex with radial k less than or equal to the station's radial from the centre of the defined region. - the station distance from the centre of the region, assuming a spherical Earth, is less than or equal to the distance kvc associated with the last vertex in the list. |
| 5.1.5.2.17a | If the station is not in the defined region, then further processing of this parameter shall terminate. |

Table 5.30: Regional multicast block bit encoding

| Description | Octet | Bit number | | | | | | | | |
|------------------------------|------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| vertex count (vc) | 1 | 1 | res | lon ₁₄ | lon ₁₃ | vc ₄ | vc ₃ | vc ₂ | vc ₁ | |
| centre latitude (lat, lat4) | 2 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ | |
| | 3 | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ | |
| centre longitude (lon, lon4) | 4 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ | |
| | 5 | lon ₄ ₄ | lon ₄ ₃ | lon ₄ ₂ | lon ₄ ₁ | lat ₄ ₄ | lat ₄ ₃ | lat ₄ ₂ | lat ₄ ₁ | |
| lower altitude (lalt) | 6 | lalt ₈ | lalt ₇ | lalt ₆ | lalt ₅ | lalt ₄ | lalt ₃ | lalt ₂ | lalt ₁ | |
| upper altitude (ualt) | 7 | ualt ₈ | ualt ₇ | ualt ₆ | ualt ₅ | ualt ₄ | ualt ₃ | ualt ₂ | ualt ₁ | |
| vertex 1 | radial 1 (r ₁) | 8 | r _{1,8} | r _{1,7} | r _{1,6} | r _{1,5} | r _{1,4} | r _{1,3} | r _{1,2} | r _{1,1} |
| | distance 1 (d ₁) | 9 | d _{1,8} | d _{1,7} | d _{1,6} | d _{1,5} | d _{1,4} | d _{1,3} | d _{1,2} | d _{1,1} |
| vertex 2 (as needed) | radial 2 (r ₂) | 10 | r _{2,8} | r _{2,7} | r _{2,6} | r _{2,5} | r _{2,4} | r _{2,3} | r _{2,2} | r _{2,1} |
| | distance 2 (d ₂) | 11 | d _{2,8} | d _{2,7} | d _{2,6} | d _{2,5} | d _{2,4} | d _{2,3} | d _{2,2} | d _{2,1} |

Table 5.31: Regional multicast block field encoding

| Subfield | Encoding | Notes |
|---|--|--|
| Vertex count (vc) | 0: vertex 1 1 to 15: vertex 2 to vertex 16 | vertex k corresponds to radial k and distance k |
| centre latitude (lat, lat4) centre longitude (lon, lon4) | lat, lon, lat4, lon4 are CPR encoded per clause 5.1.6 and per EN 302 842-2 [5] clause 5.6. | Sent using type=0, reference position is location of station transmitting this CTRL. |
| lower altitude (lalt) upper altitude (ualt) | 0: alt = -1 500 feet 1: alt = -1 250 feet 2 to 254: 250ft linear increments up to 63 500 feet 255: no upper limit | Station only matches the broadcast if its altitude (baro if available, otherwise geo) is between lower and upper altitudes. Altitude code 255 is not valid for lower altitude. |
| distance (d) | See table 5.32. | Vertex blocks are listed in monotonically increasing radial order (clockwise around centre). |
| radial (r) | due North coded as 0. Resolution is 360/256 degrees, linear | |

Table 5.32: Vertex distance encoding

| Transmitted value of distance (decimal equivalent of field) | Within region if: |
|--|---------------------|
| 0 | distance < 0,5 nmi |
| 1 | distance < 1,0 nmi |
| ↓ | ↓ |
| 59 | distance < 29,5 nmi |
| 60 | distance < 30 nmi |
| 61 | distance < 31 nmi |
| ↓ | ↓ |
| 89 | distance < 59 nmi |
| 90 | distance < 60 nmi |
| 91 | distance < 62 nmi |
| ↓ | ↓ |
| 119 | distance < 118 nmi |
| 120 | distance < 120 nmi |
| 121 | distance < 124 nmi |
| ↓ | ↓ |
| 254 | distance < 656 nmi |
| 255 | distance < 660 nmi |

| Requirement reference | |
|--|--|
| Channel management parameter reception procedures | |
| 5.1.5.2.17b | Upon receipt of a channel management CTRL command that is determined to apply to the station due to an address, broadcast or regional match, the station shall establish NTM streams. |
| 5.1.5.2.17c | If $styp = 0$, then NTM shall equal $NES \times sr$ (where sr is the result of looking up the encoded sr in table 5.28 in EN 302 842-2 [5]). |
| 5.1.5.2.17d | If $styp = 1$, then NTM shall equal $NES / (2 + sr)$. |
| 5.1.5.2.17e | If $styp=1$ and NES is not a multiple of $(2 + sr)$, then periodic streams cannot be defined according to the indicated script, the message shall be judged invalid, |
| 5.1.5.2.17f | and the entire CTRL parameter shall be discarded without further processing. |
| 5.1.5.2.17g | On receipt of a channel management parameter containing in the destination block a matching destination address, the receiving station shall update its reservation table and carry out the actions as specified in table 5.32a. |
| 5.1.5.2.17h | The m th individual offset idom shall correspond to the m th destination address matching that of the receiving station. |

Table 5.32a: Action on receipt of a channel management parameter

| Directed offset (do) | Directed timeout (dt) | Action |
|----------------------|-----------------------|--|
| 0 | any | See clause 3.7.2.3.1. |
| 1 | any | Invalid |
| $do > 1$ | $dt < 15$ | Reserve the following slots for the destination to broadcast: for j equal to 0 to 3 and k equal to 0 to $NTM - 1$, the slots equal to $truncate (do + ido_m + (k \times M1/NTM) + j \times M1)$ through $(lg + truncate (do + ido_m + (k \times M1/NTM) + j \times M1))$ after the first slot of the received burst |
| $do > 1$ | $dt = 15$ | Reserve the following slots for the destination to broadcast: for k equal to 0 to $NTM - 1$, the slots equal to $truncate (do + ido_m + (k \times M1/NTM))$ through $(lg + truncate (do + (k \times M1/NTM)))$ after the first slot of the received burst |

| Action on receipt of a channel management parameter with $do = 0$ | |
|--|--|
| 5.1.5.2.17i | On receipt of an CTRL command with the m th destination address matching that of the receiving station and with the directed offset subfield (do) equal to 0, the receiving station shall operate autonomously using the periodic broadcast procedures with the nominal periodic rate (V11) set to NTM for the next $dt \times M1$ slots, with the a/d bit set equal to 0. |
| 5.1.5.2.17j | The nominal slots shall be spaced such that the nominal slot for the k th element identified in the script occurs $truncate ((k-1) \times M1/NTM)$ slots after the nominal slot for the first element identified in the script. |
| 5.1.5.2.17k | The first transmission made in accordance with the received channel management parameter command shall occur within 4 500 slots of the start of the received command. |
| Action on receipt of a channel management parameter with $do > 1$ | |
| 5.1.5.2.17l | On receipt of an CTRL command with the m th destination address matching that of the receiving station and with the directed offset subfield (do) greater than 1, the responder station shall begin the transmission of the first script element at $T0m$ slots after the first slot of the CTRL command where $T0m = do + idom$. |
| 5.1.5.2.17m | Subsequent transmission slots shall be spaced such that the slot for the k th element identified in the script occurs $truncate ((k - 1) \times M1 / NTM)$ slots after the specified slot for the first element identified in the script. |
| 5.1.5.2.17n | The a/d bit shall be set equal to 1. |

| Setting of TV11 timer | |
|---|--|
| 5.1.5.2.17o | Upon receipt of a channel management CTRL command the station shall set the TV11 timer (see EN 302 842-2 [5], clause 5.2.10.2) equal to the value of the directed timeout (dt) subfield for each of the slots indicated in the CTRL transmission. |
| 5.1.5.2.17p | The responder station shall transmit in each of the reserved slots. |
| 5.1.5.2.17q | Each response burst shall contain the periodic broadcast reservation field with the periodic offset (po) subfield set to zero and the periodic timeout (pt) subfield set to $\min(3, TV11-1)$. |
| 5.1.5.2.17r | After each transmission, the timer TV11 shall be decremented. |
| 5.1.5.2.17s | When TV11 reaches zero, the responder shall not transmit a further response to the channel management parameter. |
| 5.1.5.2.17t | Upon cessation of directed transmissions, the responder shall resume default autonomous behaviour on the GSCs, reserving new slots as required. |
| Sharing streams | |
| 5.1.5.2.17u | If $styp = 1$, then the transmissions in successive minutes shall share streams with those script elements that are NTM (modulo NES) apart. |
| 5.1.5.2.17v | If all of the elements that share a stream are not transmitted on the same frequency, then the message shall be judged invalid |
| 5.1.5.2.17w | and the entire channel management parameter shall be discarded. |
| 5.1.5.2.17x | If all of the elements that share a stream are not of the same length, then a basic sync burst shall be transmitted for all elements in that stream. |
| 5.1.5.2.17y | If a station receives a plea request, it shall use NTM as the default reporting rate for the frequency to construct the plea response. |
| Cancellation of channel management parameter command | |
| 5.1.5.2.18 | A station shall cancel a channel management parameter command with known csid by transmitting a channel management parameter with the directed timeout subfield set to 15, the directed offset (do) set to the offset from the first slot of the cancellation channel management parameter to the first slot for which the reservation is to be cancelled, and all other subfields set to the same values as in the original channel management parameter to be cancelled. |
| 5.1.5.2.19 | A station shall cancel the set of all channel management parameter commands for a station or set of stations by transmitting a channel management parameter with $csid = 15$. |
| 5.1.5.2.19a | A receiving station that is commanded to cancel all reservations for sync burst transmission shall revert to default sync burst operations. |
| Cancellation of autonomous streams | |
| 5.1.5.2.19b | If the responder was transmitting autonomously the VSS user data for which a channel management parameter command was received, then it shall cancel its existing reservations in accordance with clause 5.2.10.5.28 in EN 302 842-2 [5], and operate in accordance with the parameters of the channel management parameter command. |
| Command set ID parameter | |
| 5.1.5.2.19c | A station that receives a channel management parameter with $csid = 0$ shall terminate any previous default sync burst operations and initiate operations in accordance with the current channel management parameter |
| 5.1.5.2.19d | A station that receives a channel management parameter with csid matching a previously-received channel management parameter will terminate any previous operations commanded by the previous channel management parameter and initiate operations in accordance with the current channel management parameter. |
| 5.1.5.2.19e | A station that does not receive a new channel management parameter block with $csid = 0$ within $(dt \times M1)$ slots, with dt as specified in the channel management parameter block with $csid = 0$, shall revert to default sync burst operations. |
| 5.1.5.2.19f | A station that receives a new channel management parameter block with $csid = 15$ shall cancel all reservations for transmission associated with the previously-received channel management commands and revert to default sync burst operations. |
| 5.1.5.2.19g | A station that does not receive a new channel management parameter block with $csid > 0$ within $(dt \times M1)$ slots, with dt as specified in the channel management parameter block with $csid > 0$, shall terminate the operations indicated in the associated channel management parameter block. |
| Transmission block encoding | |
| Pre-defined transmission block definitions | |
| 5.1.5.2.20 | The following ordered list of pre-defined transmission blocks defined in table 5.33 shall be referenced by the transmission definition index (txd) subfield (see table 5.42) in the script block when the non-standard definition (nsd) subfield (see table 5.42) is equal to zero. |
| 5.1.5.2.20a | If a station receives a channel management parameter containing a transmission definition index (txd) that it does not recognize, it shall transmit a basic sync burst. |

Table 5.33: Pre-defined transmission block definition encoding

| Encoding | Transmission definition |
|----------|--|
| 0 | Sync burst with baro altitude, no sleep parameters, basic variable field, baro alt rate. |
| 1 | Sync burst with geo altitude, no sleep parameters, basic variable field, geo alt rate. |
| 2 | Sync burst with baro altitude, no sleep parameters, full position variable field, baro alt rate. |
| 3 | Sync burst with baro altitude, no sleep parameters, aircraft data variable field. |
| 4 to 15 | Reserved. |

| Requirement reference | User-defined transmission block |
|-----------------------|--|
| 5.1.5.2.21 | Zero to fifteen user-defined transmission blocks shall be defined in the transmission block definition clause as indicated by the tc field (see table 5.27). |
| 5.1.5.2.22 | User-defined transmission blocks shall be encoded as defined in tables 5.34 and 5.36. |
| 5.1.5.2.22a | If the mobile is travelling greater than 3 069 knots (i.e., bit gs12 = 1), then the mobile shall transmit a high dynamic variable field whenever it otherwise would have transmitted a basic, full position, or high resolution variable field. |
| 5.1.5.2.22b | Whenever the mobile cannot transmit the requested information (either a new message ID, variable field, or other option was defined that the mobile does not understand or the mobile does not have the requested information), then the mobile shall transmit a sync burst with a basic variable field. |

Table 5.34: User-defined synchronization burst transmission block bit encoding

| Description | Octet | Bit number | | | | | | | |
|----------------------------|-------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ID (id) | 1 | id ₄ | id ₃ | id ₂ | id ₁ | aux ₂ | aux ₁ | b/g | 0 |
| ID-extension (if required) | 2 | id1 ₄ | id1 ₃ | id1 ₂ | id1 ₁ | id2 ₄ | id2 ₃ | id2 ₂ | id2 ₁ |

Table 5.35: User-defined synchronization burst transmission block field encoding

| Subfield | Encoding | Notes |
|------------------------------------|--|---|
| baro/geo selection (b/g) | b/g = 0: send baro b/g = 1: send geo | If the preferred altitude encoding is not available, then send the other if it is available. |
| variable field ID (and extensions) | Per table 5.1 | Octet 2 is only included when necessary |
| auxiliary selection (aux) | For those variable fields with a selection (e.g., basic, high dynamic, TCP), send selection = aux. | If the requested data is not available, but the alternate is, then send the alternate. As an example, for the basic variable field, aux=0 means send baro rate. |

Table 5.36: Non-sync burst transmission block bit encoding

| Description | Octet | Bit number | | | | | | | |
|--|-------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| message ID (mid) | 1 | mid ₈ | mid ₇ | mid ₆ | mid ₅ | mid ₄ | mid ₃ | mid ₂ | mid ₁ |
| length (len), application specific (app) | 2 | app | app | app | app | len ₄ | len ₃ | len ₂ | len ₁ |
| NOTE: Octets 3 through len+2 are all application specific. | | | | | | | | | |

Table 5.37: Non-sync burst transmission block field encoding

| Subfield | Encoding | Notes |
|---------------------------------|---------------------------------|---|
| application specific data (app) | | Defined by the application specified by the message ID |
| length (len) | 0 to 15 | Number of octets following the length (i.e. an application requiring only 4 bits of parameters would encode a 0 for length) |
| message ID (mid) | Per EN 302 842-2 [5], table 5.3 | Note that mid ₁ equals 1 for all message types except the synchronization burst - hence this bit distinguishes the format of table 5.34 from that of table 5.36. |

| Requirement reference | |
|-----------------------|---|
| | Frequency block encoding |
| 5.1.5.2.22c | Zero to seven frequencies shall be included in the frequency block clause (as indicated by the fc field). |
| 5.1.5.2.22d | A station with k available receivers shall monitor the first min(k, number of mandatory frequencies) frequencies in the frequency list. |
| 5.1.5.2.23 | A single frequency shall be encoded as defined in table 5.38 with subfield encodings as defined in table 5.39. |
| | Recommendation |
| 5.1.5.2.23a | A station with spare receivers should monitor the remaining frequencies in the list. |

Table 5.38: Frequency block bit encoding

| Description | Octet | Bit number | | | | | | | |
|-------------|-------|----------------|----------------|----------------|----------------|-----------------|-----------------|-----------------|----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| flags | 1 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| freq | 2 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |

Table 5.39: Frequency block field encoding

| Subfield | Encoding | Notes |
|--------------------|---|---|
| frequency (f) | See EN 302 842-2 [5], table 5.26 | |
| must receive (rcv) | rcv = 0: reception on the specified frequency is optional rcv = 1: reception on the specified frequency is mandatory | As the list is ordered in priority, a station receiving a list with a frequency with rcv=0 before a frequency with rcv=1 shall discard the entire channel management parameter. |
| sleep and auto | Per clauses 5.1.2.2 and 5.1.2.3 | If the respective bit is set, then the respective block is appended per clause 5.1.2.3. |

| Requirement reference | |
|-----------------------|---|
| | Parameter block definition |
| 5.1.5.2.24 | The transmitting station shall transmit all of the streams defined in the script using the parameters defined in table 5.40 with subfield encodings as defined in EN 302 842-2 [5] clause 5.2 for the appropriate variable. |
| 5.1.5.2.24a | The receiving station shall use the first pc octets from table 5.40 instead of the default values for those parameters. |
| 5.1.5.2.24b | If pc is greater than the length of table 5.40, then the receiving station shall ignore all octets beyond the length. |
| 5.1.5.2.24c | These parameters shall be ignored for directed slot operations. |

Table 5.40: Parameter block bit encoding

| Description | Octet | Bit number | | | | | | | |
|------------------|-------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| Q2a | 1 | Q2a ₈ | Q2a ₇ | Q2a ₆ | Q2a ₅ | Q2a ₄ | Q2a ₃ | Q2a ₂ | Q2a ₁ |
| Q2b | 2 | Q2b ₈ | Q2b ₇ | Q2b ₆ | Q2b ₅ | Q2b ₄ | Q2b ₃ | Q2b ₂ | Q2b ₁ |
| Q2c | 3 | Q2c ₈ | Q2c ₇ | Q2c ₆ | Q2c ₅ | Q2c ₄ | Q2c ₃ | Q2c ₂ | Q2c ₁ |
| | 4 | Q2d ₁₀ | Q2d ₉ | Q2c ₁₀ | Q2c ₉ | Q2b ₁₀ | Q2b ₉ | Q2a ₁₀ | Q2a ₉ |
| Q2d | 5 | Q2d ₈ | Q2d ₇ | Q2d ₆ | Q2d ₅ | Q2d ₄ | Q2d ₃ | Q2d ₂ | Q2d ₁ |
| Q1,Q4 | 6 | Q1 ₃ | Q1 ₂ | Q1 ₁ | Q4 ₅ | Q4 ₄ | Q4 ₃ | Q4 ₂ | Q4 ₁ |
| V12 | 7 | Q1 ₄ | V12 ₇ | V12 ₆ | V12 ₅ | V12 ₄ | V12 ₃ | V12 ₂ | V12 ₁ |
| TV11min, TV11max | 8 | TV11mi n ₄ | TV11mi n ₃ | TV11mi n ₂ | TV11m in ₁ | TV11ma x ₄ | TV11ma x ₃ | TV11ma x ₂ | TV11m ax ₁ |
| V11 | 9 | res | res | V11 ₆ | V11 ₅ | V11 ₄ | V11 ₃ | V11 ₂ | V11 ₁ |
| VS2 (nmi) | 10 | res | res | VS2 ₆ | VS2 ₅ | VS2 ₄ | VS2 ₃ | VS2 ₂ | VS2 ₁ |
| VS4 | 11 | res | VS4 ₇ | VS4 ₆ | VS4 ₅ | VS4 ₄ | VS4 ₃ | VS4 ₂ | VS4 ₁ |

| Requirement reference | |
|-----------------------|--|
| | Script block definition |
| 5.1.5.2.25 | The channel management CTRL parameter script shall be encoded as defined in table 5.41 with subfield encoding as defined in table 5.42. |
| 5.1.5.2.26 | The script block shall contain one or more script elements. |
| 5.1.5.2.27 | The two types of script elements shall be as shown in table 5.41. |
| 5.1.5.2.28 | A transmitting station shall not transmit a script block with two consecutive octets having bits 6 to 8 equal to "111". |
| 5.1.5.2.29 | A receiving station shall discard a channel management CTRL command containing a script block with two consecutive octets having bits 6 to 8 equal to "111". |

Table 5.41: Script block bit encoding

| Description | Octet | Bit number | | | | | | | |
|--|-------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| freq index (frq), transmission definition index (txd) | 1 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| | | | | | or | | | | |
| repeat count (rpt) | 1 | 1 | 1 | 1 | rpt ₅ | rpt ₄ | rpt ₃ | rpt ₂ | rpt ₁ |
| freq index (frq), non-standard definition (nsd), transmission definition index (txd) | 2 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |

Table 5.42: Script block field encoding

| Subfield | Encoding | Notes |
|-------------------------------------|--|---|
| frequency index (frq) | 0 to 6: integer index into freq block 7: repeat flag (second octet with freq index follows) | The offset from the beginning of the frequency list (an encoding of 0 refers to the first element in the frequency block). |
| Non-standard definition (nsd) | nsd = 0: txd encoded as defined in table 5.33. nsd = 1: txd indicates offset from beginning of transmission definition list (see clause 5.1.5.2.18) | |
| Transmission definition index (txd) | 0 to 15 | The offset from the beginning of the transmission definition list (an encoding of 0 refers to the first element in the list). |
| Repeat count (rpt) | 3 (encoded as 0) to 34 | The number of times to repeat the frequency and (nsd, txd) listed in octet 2. |

5.1.6 Definitions for Compact Position Reporting

5.1.6.1 Information Field Offset Encoding

| Requirement reference | |
|-----------------------|--|
| | General |
| 5.1.6.1.1 | Given a position and its fixed data field encoding, a high resolution offset encoding of size <i>bits</i> together with an additional bit indicating the sign of the offset shall be computed as defined below. |
| 5.1.6.1.2 | The offset shall be encoded in the sub-fields <i>lat4</i> , <i>lon4</i> (for <i>bits</i> = 3), <i>lat6</i> , <i>lon6</i> (for <i>bits</i> = 5) or <i>lat8</i> , <i>lon8</i> (for <i>bits</i> = 7) dependent on which synchronization burst variable part is to be transmitted. |
| | Input parameters |
| 5.1.6.1.3 | The input parameters used for information field offset encoding shall be defined as follows: <i>clat_{in}</i> = latitude to be encoded. <i>clon_{in}</i> = longitude to be encoded. <i>lat</i> = the CPR encoded latitude. <i>lon</i> = the CPR encoded longitude. <i>type</i> = type of CPR (odd or even). <i>bits</i> = number of bits in magnitude of the offset. |
| | Calculations |
| | Supporting functions |
| 5.1.6.1.4 | The supporting function for calculating the information field offset shall be as follows: $\text{diff}(pos_1, pos_2) = \begin{cases} pos_1 - pos_2 - MAX_C - 1 & \text{if } pos_1 > pos_2 + MAX_C / 2 \\ pos_1 - pos_2 + MAX_C + 1 & \text{if } pos_2 > pos_1 + MAX_C / 2 \\ pos_1 - pos_2 & \text{else} \end{cases}$ $\text{sign}(pos_1, pos_2) = \begin{cases} 1 & \text{if } \text{diff}(pos_1, pos_2) \geq 0 \\ 0 & \text{else} \end{cases}$ |
| | Latitude |
| 5.1.6.1.5 | The encoded fixed latitude offset shall be calculated as follows: $lat = \text{enc}_{lat}(clat_{in}, type)$ $clat_{dec} = \text{dec}_{lat}(clat_{in}, lat, lat, type)$ $tmp_7 = \frac{MAX_C}{2 \cdot nz(type) \cdot MAX_T^{lat} \cdot (2^{bits} - 1)}$ $\text{offset}_{enc}^{lat}(clat_{in}, clat_{dec}, bits, type) = \frac{ \text{diff}(clat_{in}, clat_{dec}) + \frac{tmp_7}{2}}{tmp_7}$ $lat4(bits\ 1\ to\ 3) = \text{offset}_{enc}^{lat}(clat_{in}, clat_{dec}, 3, type)$ $lat6(bits\ 1\ to\ 5) = \text{offset}_{enc}^{lat}(clat_{in}, clat_{dec}, 5, type)$ $lat8(bits\ 1\ to\ 7) = \text{offset}_{enc}^{lat}(clat_{in}, clat_{dec}, 7, type)$ $lat4(bit\ 4) = lat6(bit\ 6) = lat8(bit\ 8) = \text{sign}(clat_{in}, clat_{dec})$ |

| Requirement reference | Longitude |
|-----------------------|---|
| 5.1.6.1.6 | <p>The encoded fixed longitude offset shall be calculated as follows:</p> $lon = enc_{lon}(clat_{dec}, clon_{in}, type)$ $clon_{dec} = dec_{lon}(clat_{dec}, clon_{in}, lon, lon, type)$ $tmp_8 = \frac{MAX_C}{2 \cdot nl(clat_{dec}, type) \cdot MAX_T^{lon} \cdot (2^{bits} - 1)}$ $offset_{enc}^{lon}(clat_{dec}, clon_{in}, clon_{dec}, bits, type) = \frac{ diff(clon_{in}, clon_{dec}) + \frac{tmp_8}{2}}{tmp_8}$ $lon4(bits1to3) = offset_{enc}^{lon}(clat_{dec}, clon_{in}, clon_{dec}, 3, type)$ $lon6(bits1to5) = offset_{enc}^{lon}(clat_{dec}, clon_{in}, clon_{dec}, 5, type)$ $lon8(bits1to7) = offset_{enc}^{lon}(clat_{dec}, clon_{in}, clon_{dec}, 7, type)$ $lon4(bit4) = lon6(bit6) = lon8(bit8) = sign(clon_{in}, clon_{dec})$ |

5.1.6.2 Information Field Offset Decoding

| Requirement reference | |
|-----------------------|--|
| | General |
| 5.1.6.2.1 | To decode the information field latitude and longitude offsets, the offset values defined below shall be added to the position from the fixed data field position decoding (either local or global) to determine the actual target position. |
| | Input parameters |
| 5.1.6.2.2 | The input parameters used for information field offset decoding shall be defined as follows: <i>lat4, lat6, lat8</i> = received latitude offset. <i>lon4, lon6, lon8</i> = received longitude offset. <i>type</i> = type of encoding (odd or even). <i>clatdec</i> = the target's decoded latitude |
| | Calculations |
| | Latitude |
| 5.1.6.2.3 | The decoded fixed latitude offset shall be calculated as follows: $bits = \begin{cases} 3 & \text{if } lat4, lon4 \text{ provided as input} \\ 5 & \text{if } lat6, lon6 \text{ provided as input} \\ 7 & \text{if } lat8, lon8 \text{ provided as input} \end{cases}$ $lat_{offs} = \begin{cases} lat4(bits\ 1\ to\ 3) & \text{if } lat4, lon4 \text{ provided as input} \\ lat6(bits\ 1\ to\ 5) & \text{if } lat6, lon6 \text{ provided as input} \\ lat8(bits\ 1\ to\ 7) & \text{if } lat8, lon8 \text{ provided as input} \end{cases}$ $s_{lat} = \begin{cases} lat4(bit\ 4) & \text{if } lat4, lon4 \text{ provided as input} \\ lat6(bit\ 6) & \text{if } lat6, lon6 \text{ provided as input} \\ lat8(bit\ 8) & \text{if } lat8, lon8 \text{ provided as input} \end{cases}$ $tmp_9 = \frac{MAX_C}{2 \cdot nz(type) \cdot MAX_T^{lat} \cdot (2^{bits} - 1)}$ $offset_{dec}^{lat}(lat_{offs}, s_{lat}, bits, type) = \begin{cases} lat_{offs} \cdot tmp_7 & \text{if } s_{lat} = 1 \\ -lat_{offs} \cdot tmp_7 & \text{if } s_{lat} = 0 \end{cases}$ |
| | Longitude |
| 5.1.6.2.4 | The decoded fixed longitude offset shall be calculated as follows: $lon_{offs} = \begin{cases} lon4(bits\ 1\ to\ 3) & \text{if } lat4, lon4 \text{ provided as input} \\ lon6(bits\ 1\ to\ 5) & \text{if } lat6, lon6 \text{ provided as input} \\ lon8(bits\ 1\ to\ 7) & \text{if } lat8, lon8 \text{ provided as input} \end{cases}$ $s_{lon} = \begin{cases} lon4(bit\ 4) & \text{if } lat4, lon4 \text{ provided as input} \\ lon6(bit\ 6) & \text{if } lat6, lon6 \text{ provided as input} \\ lon8(bit\ 8) & \text{if } lat8, lon8 \text{ provided as input} \end{cases}$ $tmp_{10} = \frac{MAX_C}{2 \cdot nl(clat_{dec}, type) \cdot MAX_T^{lon} \cdot (2^{bits} - 1)}$ $offset_{dec}^{lon}(clat_{dec}, lon_{offs}, s_{lon}, bits, type) = \begin{cases} lon_{offs} \cdot tmp_8 & \text{if } s_{lon} = 1 \\ -lon_{offs} \cdot tmp_8 & \text{if } s_{lon} = 0 \end{cases}$ |

5.1.6.3 Patch ID Encoding

| Requirement reference | |
|-----------------------|--|
| | General |
| 5.1.6.3.1 | When a station sends an unambiguous global position in a single message, then it shall transmit the value of $enc_{patch}()$ as the patch ID (pid). |
| | Input parameters |
| 5.1.6.3.2 | The input parameters used for patch ID encoding shall be defined as follows: $clat_{in}$ = latitude to encode. $clon_{in}$ = longitude to encode. $type$ = type of encoding (odd or even). |
| | Calculations |
| 5.1.6.3.3 | The encoded patch ID shall be calculated as follows: $lat_p = \begin{cases} \frac{clat_{in}}{dlat(type)} & \text{if } 0 \leq clat_{in} \leq \frac{MAX_c}{4} \\ \frac{clat_{in}}{dlat(type)} - 16 & \text{if } clat_{in} > \frac{MAX_c}{4} \end{cases}$ $lat = enc_{lat}(clat_{in}, type)$ $clat_{dec} = dec_{lat}(clat_{in}, lat, lat, type)$ $lon_p = \frac{clon_{in}}{dlon(clat_{dec}, type)}$ $pid = enc_{patch}(lat_p, lon_p) = 36 \cdot lat_p + lon_p$ $offset_{dec}^{lat}(lat_{offs}, s_{lat}, bits, type) = \begin{cases} lat_{offs} \cdot tmp_7 & \text{if } s_{lat} = 1 \\ -lat_{offs} \cdot tmp_7 & \text{if } s_{lat} = 0 \end{cases}$ |

5.1.6.4 Patch Id Decoding

| Requirement reference | |
|-----------------------|---|
| | General |
| 5.1.6.4.1 | The patch ID decoding shall use the received patch ID , lat and lon subfields and decode this to a position in latitude and longitude, which is globally unambiguous. |
| | Input parameters |
| 5.1.6.4.2 | The input parameters used for patch ID decoding shall be defined as follows: <i>lat</i> = CPR latitude to decode. <i>lon</i> = CPR longitude to be decode. <i>type</i> = type of encoding (odd or even). <i>pid</i> = encoded patch id |
| | Calculations |
| | Latitude |
| 5.1.6.4.3 | The decoded latitude component of the patch ID shall be calculated as follows: $lat_p = \begin{cases} \frac{pid}{36} & \text{if } 0 \leq pid \leq 359 \\ \frac{pid}{36} + 16 & \text{if } pid > 359 \end{cases}$ $fullDec_{lat}(lat, lat_p, type) = \frac{\left(\left(\frac{MAX_C}{MAX_T^{lat}} \right) \cdot lat \right)}{nz(type)} + dlat(type) \cdot lat_p$ |
| | Longitude |
| 5.1.6.4.4 | The patch ID longitude component shall be calculated as follows: $lon_p = \text{mod}(pid, 36)$ $clat_{dec} = fullDec_{lat}(lat, lat_p, type)$ $fullDec_{lon}(clat_{dec}, lon, lon_p, type) = \frac{\left(\left(\frac{MAX_C}{MAX_T^{lon}} \right) \cdot lon \right)}{nl(clat_{dec}, type)} + dlon(clat_{dec}, type) \cdot lon_p$ |

5.1.7 Encoding of UDATA (udid)

| Requirement reference | |
|-----------------------|---|
| 5.1.7.1 | The UDATA ID (udid) field shall be encoded as defined in table 5.42a. |

Table 5.42a: Encoding of UDATA ID (udid)

| Service | UDATA ID (udid) |
|----------|----------------------|
| TIS-B | 00000bin |
| FIS-B | 00001bin |
| Reserved | 00010bin |
| GNS-B | 00011bin |
| Reserved | 00100bin to 11101bin |

5.2 Requirements for TIS-B

5.2.1 Traffic Information Volume (TIV)

| Requirement reference | |
|--|---|
| 5.2.1.1 | Void |
| 5.2.1.2 | Each TIV from a ground station shall be identified by a TIV identity code (TIV ID). |
| 5.2.1.3 | Each TIV shall be uniquely determined by the combination of the ground station 24-bit ICAO address (sent in every VDL Mode 4 burst) and the TIV ID. |
| 5.2.1.4 | A TIV shall be either a polygon shape with a maximum of 10 vertices, or a circle with a radius defined by one vertex (see notes 1 and 2). |
| 5.2.1.5 | The TIV polygon shall be two-dimensional and not have any crossing borders. |
| 5.2.1.6 | No vertices shall be the same for a TIV. |
| 5.2.1.7 | Each TIV shall have a minimum altitude limit and a greater maximum altitude limit (see note 3). |
| NOTE 1: A polygon TIV may be concave or convex. | |
| NOTE 2: Other TIV shapes may be defined in future by encoding only one or two vertices. | |
| NOTE 3: The minimum and maximum altitude limits should not be the same. The altitude limit applies in the whole TIV. | |

5.2.2 Void

5.2.3 Message format

| Requirement reference | |
|-----------------------|---|
| 5.2.3.1 | TIS-B messages shall be contained in a DLS UDATA DLPDU burst of type UINFO as described in clause 5.3 of EN 302 842-2 [5]. |
| 5.2.3.2 | A station shall recognize a TIS-B service by the UDATA ID (udid) field being set to "00000" (see clause 5.3.1.3 of EN 302 842-2 [5]). |
| 5.2.3.3 | TIS-B messages shall be contained in the information field of a VDL Mode 4 burst, with the first message starting in bit 1 of octet 6. |
| 5.2.3.4 | Where more than one TIS-B message is sent in one TIS-B burst, each additional TIS-B message shall start in bit 1 of the octet following the last octet occupied by the previous TIS-B message. |
| 5.2.3.5 | Where more than one TIS-B message is contained in one TIS-B burst, if the previous TIS-B message does not finish at an octet boundary, up to 7 bits of zeros shall be included to complete the octet, prior to the start of the subsequent TIS-B message. |
| 5.2.3.6 | A station shall recognize each TIS-B message by the TIS-B message identifier (tmi), as defined in table 5.43. |
| 5.2.3.7 | To indicate a TIS-B message identifier greater than 14, bits 1 to 4 of octet n + 1 shall be set to 15, and a further 8 bits of tmi1 shall be inserted starting in bit 5 of octet n + 1 and ending in bit 4 of octet n + 2, so that $tmi = 15 + tmi1$. |

| Requirement reference | |
|-----------------------|--|
| 5.2.3.8 | <p>A station shall be capable of decoding TIS-B bursts appended with the following reservation fields:</p> <ul style="list-style-type: none"> - a reservation ID (rid) equal to 1, with the burst appended by a null reservation field as defined in clause 5.9 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0 and an extended reservation ID (erid) equal to 00000binary, with the burst appended by a response reservation field as defined in clause 5.2.9 of EN 302 842-2 [5], with address type field equal to 7; - a reservation ID (rid) equal to 1, with the burst appended by a periodic broadcast reservation field as defined in clause 5.2.10 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0, with the burst appended by an incremental broadcast reservation field as defined in clause 5.2.11 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 1, with the burst appended by a combined periodic broadcast and incremental broadcast reservation field as defined in clause 5.2.12 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0, with the burst appended by a unicast reservation field with sdf=1 as defined in clause 5.2.14 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0, with the burst appended by a second frame reservation field as defined in clause 5.2.17 of EN 302 842-2 [5]; or - a reservation ID (rid) equal to 0, with the burst appended by a superframe reservation field as defined in clause 5.2.17 of EN 302 842-2 [5]. |

Table 5.43: TIS-B message identifier encoding

| Message | TIS-B Message ID Encoded value | 8-bit Message ID Extension (tmi1) Encoded value |
|---|--------------------------------|---|
| Management message | 0 | not present |
| Aircraft target (airborne TIV) | 1 | not present |
| Aircraft target (ground TIV) | 2 | not present |
| Ground vehicle target (ground TIV) | 3 | not present |
| Reserved for future use | 4 to 14 | not present |
| Available for future use with 8-bit extension | 15 | 0 to 255 |

5.2.4 Management message

| Requirement reference | |
|-----------------------|---|
| 5.2.4.1 | A station shall be capable of decoding a management message as defined in table 5.44. |
| 5.2.4.2 | Fields in the management message shall be decoded as defined in table 5.45. |
| 5.2.4.3 | The latitude/longitude coordinates of a fixed virtual reference position shall be received in the management message, given in absolute WGS-84 coordinates. |
| 5.2.4.4 | The latitude/longitude positions of the TIV vertices received in the management message shall be decoded as an offset from the reference position. |
| 5.2.4.5 | All latitude/longitude positions in the target messages shall be encoded as an offset from the reference position. |

Table 5.44: Management message bit encoding

| Description | Octet | Bit number | | | | | | | |
|---|-----------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| udid, ucd | 5 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| TIS-B message ID (tmi) version number (vers) | n | vers ₄ | vers ₃ | vers ₂ | vers ₁ | tmi ₄ | tmi ₃ | tmi ₂ | tmi ₁ |
| TIV ID (tivid) update period (up) | n + 1 | up ₅ | up ₄ | up ₃ | up ₂ | up ₁ | tivid ₃ | tivid ₂ | tivid ₁ |
| TIV status (tivs) TIS-B target resolution (tacc) | n + 2 | lat ₁ | tacc ₄ | tacc ₃ | tacc ₂ | tacc ₁ | tivs ₂ | tivs ₁ | up ₆ |
| reference point latitude (lat) | n + 3 | lat ₉ | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ |
| reference point longitude (long) | n + 4 | long ₃ | long ₂ | long ₁ | lat ₁₄ | lat ₁₃ | lat ₁₂ | lat ₁₁ | lat ₁₀ |
| | n + 5 | long ₁₁ | long ₁₀ | long ₉ | long ₈ | long ₇ | long ₆ | long ₅ | long ₄ |
| lower barometric altitude (lbar) | n + 6 | lbar ₄ | lbar ₃ | lbar ₂ | lbar ₁ | long ₁₅ | long ₁₄ | long ₁₃ | long ₁₂ |
| upper barometric altitude (ubar) | n + 7 | ubar ₄ | ubar ₃ | ubar ₂ | ubar ₁ | lbar ₈ | lbar ₇ | lbar ₆ | lbar ₅ |
| number of vertices (nvert) | n + 8 | nvert ₄ | nvert ₃ | nvert ₂ | nvert ₁ | ubar ₈ | ubar ₇ | ubar ₆ | ubar ₅ |
| vertex latitude (vlat(1-k)) | n + 9 | vlat ₁₈ | vlat ₁₇ | vlat ₁₆ | vlat ₁₅ | vlat ₁₄ | vlat ₁₃ | vlat ₁₂ | vlat ₁₁ |
| vertex longitude (vlg(1 -k)) | n + 10 | vlg ₁₈ | vlg ₁₇ | vlg ₁₆ | vlg ₁₅ | vlg ₁₄ | vlg ₁₃ | vlg ₁₂ | vlg ₁₁ |
| | | | | | | | | | |
| | n + 2k +7 | vlatk ₈ | vlatk ₇ | vlatk ₆ | vlatk ₅ | vlatk ₄ | vlatk ₃ | vlatk ₂ | vlatk ₁ |
| | n + 2k +8 | vlgk ₈ | vlgk ₇ | vlgk ₆ | vlgk ₅ | vlgk ₄ | vlgk ₃ | vlgk ₂ | vlgk ₁ |

..... [Denotes variable length field]

Table 5.45: Management message field encoding

| Data field | Definition | Encoded Value |
|---------------------------|--|--|
| TIS-B message ID | See table 5.43 | 0 |
| TIS-B service version | TIS-B as defined in the present document Unused | 0 1 to 15 |
| TIV ID | 1 to 8 | 0 to 7 |
| Update period | 0,5 s 1,0 s 2,0 s 3,0 s ↓ 60 s Unused | 0 1 2 3 ↓ 60 61 to 63 |
| TIV status | OK Not OK Overloaded Unused | 0 1 2 3 |
| TIS-B target resolution | See position Navigation Integrity Category (NIC) encoding in table 5.59 in clause 5.4.2.3 of EN 302 842-2 [5] | |
| Reference point latitude | Encoded reference point latitude = (latitude) × 60 + 5 400, where latitude has an increment of 1/60 th of a degree and range between -90 degrees and +90 degrees Unused | 0 to 10 800 10 801 to 16 383 |
| Reference point longitude | Encoded reference point longitude = (longitude) × 60 + 10 800, where longitude has an increment of 1/60 th of a degree and range between -180 degrees and +180 degrees Unused | 0 to 21 600 21 601 to 32 767 |
| Lower barometric altitude | -1 500 feet -1 250 feet 250 ft linear increments up to 63 500 feet Encoding of 255 is not allowed | 0 1 2 to 254 |
| Upper barometric altitude | -1 500 feet -1 250 feet 250 ft linear increments up to 63 500 feet No upper limit | 0 1 2 to 254 255 |
| Number of vertices | Defines a specific TIV shape or the number of polygon vertices: Circle with centre at reference position and radius specified by one vertex position (comprising vlat1 and vlg1) No. of polygon vertices (3 to 10) Unused (see note) Other method of TIV shape definition or an indication of additional number of vertices is defined in an additional octet in row n + 9 | 0 1 to 8 9 to 14 15 |
| TIV vertex latitude | Encoded as an offset from the reference position (see clause 5.2.8) | |
| TIV vertex longitude | Encoded as an offset from the reference position (see clause 5.2.8) | |
| NOTE: | Encodings 9 to 14 could be used in future to encode a number of fixed shapes to save defining all vertices, or to encode further vertices, depending on future requirements. To encode fixed shapes, one or two vertices would be sufficient to define the complete TIV: e.g. square, regular hexagons (two different orientations), rectangle, "stretched" hexagons (two different orientations). | |

5.2.5 Aircraft target messages (airborne TIV)

| Requirement reference | |
|-----------------------|--|
| 5.2.5.1 | A station shall be capable of decoding an aircraft target message (airborne TIV) as defined in table 5.46. |
| 5.2.5.2 | The aircraft target message (airborne TIV) shall consist of a fixed and an optional field. |
| 5.2.5.3 | The optional field shall consist of flight ID type, flight ID and aircraft category. |
| 5.2.5.4 | The flight ID shall be either callsign or registration marking. |
| 5.2.5.5 | Void |
| 5.2.5.6 | Void |
| 5.2.5.7 | Fields in the aircraft target message (airborne TIV) shall be encoded as defined in table 5.47. |

Table 5.46: Aircraft target message (airborne TIV) bit encoding (with callsign as an example flight ID)

| Description | Octet | Bit number | | | | | | | |
|---|--------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| udid, ucd | 5 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| TIS-B message ID (tmi) TIV ID (tivid) target identifier flag (tflg) | n | tflg | tivid ₃ | tivid ₂ | tivid ₁ | tmi ₄ | tmi ₃ | tmi ₂ | tmi ₁ |
| radar/fusion flag (rflg) ADS-B fault flag (adsff) flight ID flag (fidflg) | n + 1 | tid ₅ | tid ₄ | tid ₃ | tid ₂ | tid ₁ | fidflg | adsff | rflg |
| target identifier (tid) | n + 2 | tid ₁₃ | tid ₁₂ | tid ₁₁ | tid ₁₀ | tid ₉ | tid ₈ | tid ₇ | tid ₆ |
| | n + 3 | tid ₂₁ | tid ₂₀ | tid ₁₉ | tid ₁₈ | tid ₁₇ | tid ₁₆ | tid ₁₅ | tid ₁₄ |
| latitude (rlat) | n + 4 | rlat ₅ | rlat ₄ | rlat ₃ | rlat ₂ | rlat ₁ | tid ₂₄ | tid ₂₃ | tid ₂₂ |
| | n + 5 | rlat ₁₃ | rlat ₁₂ | rlat ₁₁ | rlat ₁₀ | rlat ₉ | rlat ₈ | rlat ₇ | rlat ₆ |
| longitude (rlg) | n + 6 | rlg ₅ | rlg ₄ | rlg ₃ | rlg ₂ | rlg ₁ | rlat ₁₆ | rlat ₁₅ | rlat ₁₄ |
| | n + 7 | rlg ₁₃ | rlg ₁₂ | rlg ₁₁ | rlg ₁₀ | rlg ₉ | rlg ₈ | rlg ₇ | rlg ₆ |
| ground track (gtk) | n + 8 | gtk ₅ | gtk ₄ | gtk ₃ | gtk ₂ | gtk ₁ | rlg ₁₆ | rlg ₁₅ | rlg ₁₄ |
| barometric altitude (bar) | n + 9 | bar ₅ | bar ₄ | bar ₃ | bar ₂ | bar ₁ | gtk ₈ | gtk ₇ | gtk ₆ |
| ground speed (gsp) | n + 10 | gsp ₁ | bar ₁₂ | bar ₁₁ | bar ₁₀ | bar ₉ | bar ₈ | bar ₇ | bar ₆ |
| | n + 11 | gsp ₉ | gsp ₈ | gsp ₇ | gsp ₆ | gsp ₅ | gsp ₄ | gsp ₃ | gsp ₂ |
| time stamp (tag) | n + 12 | tag ₆ | tag ₅ | tag ₄ | tag ₃ | tag ₂ | tag ₁ | gsp ₁₁ | gsp ₁₀ |
| flight ID type (fidty) | n + 13 | call ₇ | call ₆ | call ₅ | call ₄ | call ₃ | call ₂ | call ₁ | fidty |
| callsign (call)(example) | n + 14 | call ₁₅ | call ₁₄ | call ₁₃ | call ₁₂ | call ₁₁ | call ₁₀ | call ₉ | call ₈ |
| | n + 15 | call ₂₃ | call ₂₂ | call ₂₁ | call ₂₀ | call ₁₉ | call ₁₈ | call ₁₇ | call ₁₆ |
| | n + 16 | call ₃₁ | call ₃₀ | call ₂₉ | call ₂₈ | call ₂₇ | call ₂₆ | call ₂₅ | call ₂₄ |
| | n + 17 | call ₃₉ | call ₃₈ | call ₃₇ | call ₃₆ | call ₃₅ | call ₃₄ | call ₃₃ | call ₃₂ |
| aircraft category (cat) | n + 18 | cat ₅ | cat ₄ | cat ₃ | cat ₂ | cat ₁ | call ₄₂ | call ₄₁ | call ₄₀ |

Table 5.47: Aircraft target message (airborne TIV) field encoding

| Data field | Definition | Encoded Value |
|------------------------------------|--|------------------------------------|
| TIS-B message ID | See table 5.43 | 1 |
| TIV ID | Identifies TIV for ground stations configured to transmit more than one TIV | 0 to 7 |
| Target identifier flag | Aircraft address (24-bit ICAO address) Special identifier (see note) | 0 1 |
| Radar fusion flag | TIS-B target based only on radar or multilateration data TIS-B target includes ADS-B as a data source | 0 1 |
| ADS-B fault flag | ADS-B data corresponds with TIS-B data TIS-B data should be used in preference to ADS-B data | 0 1 |
| Flight ID Flag | Optional information is not present Optional information is present | 0 1 |
| Target identifier | Either aircraft address (24-bit ICAO address) or a special identifier | |
| Latitude | Encoded as an offset from the reference position (see clause 5.2.8) | |
| Longitude | Encoded as an offset from the reference position (see clause 5.2.8) | |
| Ground track | See ground track encoding in table 5.12 | |
| Barometric altitude | See altitude encoding in table 5.52 of EN 302 842-2 [5] | |
| Ground speed | See ground speed encoding in table 5.14 | |
| Time stamp (relative) | time reference 200 ms after time reference 400 ms after time reference ↓ 11,8 s after time reference Unused | 0 1 2 ↓ 59 60 to 63 |
| Flight ID type | Callsign Registration marking (tail number) | 0 1 |
| Callsign (See table 5.11) | Callsign is left justified. Valid characters are A - Z, 0 - 9 and null: Assigned A - Z = 0 - 25, 0 - 9 = 26 - 35, null = 36. Callsign is an eight character string "c ₁ , c ₂ , c ₃ , c ₄ , c ₅ , c ₆ , c ₇ , c ₈ ", where: $csl = c_1 37^3 + c_2 37^2 + c_3 37 + c_4$ $csr = c_5 37^3 + c_6 37^2 + c_7 37 + c_8$ csl = left hand part of the callsign, and csr = right hand part | |
| Registration marking (tail number) | Encoding as for callsign | |
| Aircraft category | See aircraft category encoding in table 5.16 | |
| NOTE: | This is code defined by the ground system. It is unique in the TIV at any time. The first part of the identification could be the Mode A code or the tracker identification for that target followed by a unique number allocated by the ground system. | |

5.2.6 Aircraft target messages (ground TIV)

| Requirement reference | |
|-----------------------|---|
| 5.2.6.1 | A station shall be capable of decoding an aircraft target message (ground TIV) defined in table 5.48. |
| 5.2.6.2 | The aircraft target message (ground TIV) shall consist of a fixed and an optional field. |
| 5.2.6.3 | The optional field shall consist of flight ID type, flight ID and aircraft category. |
| 5.2.6.4 | The flight ID shall be either callsign or registration marking. |
| 5.2.6.5 | Void. |
| 5.2.6.6 | Void. |
| 5.2.6.7 | Fields in the aircraft target messages (ground TIV) report shall be decoded as defined in table 5.49. |

Table 5.48: Aircraft target message (ground TIV) bit encoding (with callsign as an example flight ID)

| Description | Octet | Bit number | | | | | | | |
|--|--------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| udid, ucd | 5 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| TIS-B message ID (tmi) TIV ID (tivid) target identifier flag (tflg) | n | tflg | tivid ₃ | tivid ₂ | tivid ₁ | tmi ₄ | tmi ₃ | tmi ₂ | tmi ₁ |
| radar/fusion flag (rflg) ADS-B fault flag (adsff) flight ID flag (fidflg) target identifier (tid) | n + 1 | tid ₅ | tid ₄ | tid ₃ | tid ₂ | tid ₁ | fidflg | adsff | rflg |
| | n + 2 | tid ₁₃ | tid ₁₂ | tid ₁₁ | tid ₁₀ | tid ₉ | tid ₈ | tid ₇ | tid ₆ |
| | n + 3 | tid ₂₁ | tid ₂₀ | tid ₁₉ | tid ₁₈ | tid ₁₇ | tid ₁₆ | tid ₁₅ | tid ₁₄ |
| latitude (rlat) | n + 4 | rlat ₅ | rlat ₄ | rlat ₃ | rlat ₂ | rlat ₁ | tid ₂₄ | tid ₂₃ | tid ₂₂ |
| | n + 5 | rlat ₁₃ | rlat ₁₂ | rlat ₁₁ | rlat ₁₀ | rlat ₉ | rlat ₈ | rlat ₇ | rlat ₆ |
| longitude (rlg) | n + 6 | rlg ₁ | rlg ₂₀ | rlg ₁₉ | rlg ₁₈ | rlg ₁₇ | rlg ₁₆ | rlg ₁₅ | rlg ₁₄ |
| | n + 7 | rlg ₉ | rlg ₈ | rlg ₇ | rlg ₆ | rlg ₅ | rlg ₄ | rlg ₃ | rlg ₂ |
| | n + 8 | rlg ₁₇ | rlg ₁₆ | rlg ₁₅ | rlg ₁₄ | rlg ₁₃ | rlg ₁₂ | rlg ₁₁ | rlg ₁₀ |
| ground track (gtk) | n + 9 | gtk ₅ | gtk ₄ | gtk ₃ | gtk ₂ | gtk ₁ | rlg ₂₀ | rlg ₁₉ | rlg ₁₈ |
| ground speed (gsp) | n + 10 | gsp ₅ | gsp ₄ | gsp ₃ | gsp ₂ | gsp ₁ | gtk ₈ | gtk ₇ | gtk ₆ |
| time stamp (tag) | n + 11 | tag ₂ | tag ₁ | gsp ₁₁ | gsp ₁₀ | gsp ₉ | gsp ₈ | gsp ₇ | gsp ₆ |
| flight ID type (fidty) | n + 12 | res | res | res | res | tag ₆ | tag ₅ | tag ₄ | tag ₃ |
| callsign (call) | n + 13 | call ₇ | call ₆ | call ₅ | call ₄ | call ₃ | call ₂ | call ₁ | fidty |
| | n + 14 | call ₁₅ | call ₁₄ | call ₁₃ | call ₁₂ | call ₁₁ | call ₁₀ | call ₉ | call ₈ |
| | n + 15 | call ₂₃ | call ₂₂ | call ₂₁ | call ₂₀ | call ₁₉ | call ₁₈ | call ₁₇ | call ₁₆ |
| | n + 16 | call ₃₁ | call ₃₀ | call ₂₉ | call ₂₈ | call ₂₇ | call ₂₆ | call ₂₅ | call ₂₄ |
| | n + 17 | call ₃₉ | call ₃₈ | call ₃₇ | call ₃₆ | call ₃₅ | call ₃₄ | call ₃₃ | call ₃₂ |
| aircraft category (cat) | n + 18 | cat ₅ | cat ₄ | cat ₃ | cat ₂ | cat ₁ | call ₄₂ | call ₄₁ | call ₄₀ |

Table 5.49: Aircraft target message (ground TIV) field encoding

| Data field | Definition | Encoded Value |
|------------------------------------|---|---------------|
| TIS-B message ID | See table 5.43 | 2 |
| Target identifier | Either aircraft address (24-bit ICAO address) or a special identifier | |
| TIV ID | Identifies TIV for ground stations configured to transmit more than one TIV | 0 to 7 |
| Target identifier flag | Aircraft address (24-bit ICAO address) Special identifier (see note) | 0 1 |
| Radar fusion flag | TIS-B target based only on radar or multilateration data TIS-B target includes ADS-B as a data source | 0 1 |
| ADS-B fault flag | ADS-B data corresponds with TIS-B data TIS-B data should be used in preference to ADS-B data | 0 1 |
| Flight ID Flag | Optional information is not present Optional information is present | 0 1 |
| Target identifier | Either aircraft address (24-bit ICAO address) or a special identifier | |
| Latitude | Encoded as an offset from the reference position (see clause 5.2.8) | |
| Longitude | Encoded as an offset from the reference position (see clause 5.2.8) | |
| Ground track | See ground track encoding in table 5.12 | |
| Barometric altitude | See altitude encoding in table 5.52 in clause 5.4.2.3 of EN 302 842-2 [5] | |
| Altitude resolution flag | Altitude encoded to 25 ft resolution Altitude encoded to 100 ft resolution | 0 1 |
| Ground speed | See ground speed encoding in table 5.14 in clause 5.1.1 | |
| Ground track | See ground track encoding in table 5.12 in clause 5.1.1 | |
| Time Stamp (relative) | See table 5.4.7 | |
| Flight ID type | See table 5.4.7 | |
| Callsign | See table 5.4.7 | |
| (See table 5.11) | | |
| Registration marking (tail number) | See table 5.4.7 | |
| Aircraft category | See table 5.4.7 | |
| NOTE: | This is code defined by the ground system. It is unique in the TIV at any time. The first part of the identification could be the Mode A code or the tracker identification for that target followed by a unique number allocated by the ground system. | |

5.2.7 Ground vehicle target messages (ground TIV)

| Requirement reference | |
|-----------------------|--|
| 5.2.7.1 | A station will be capable of decoding a ground vehicle target message (ground TIV) as defined in table 5.50. |
| 5.2.7.2 | Fields in the ground vehicle target message (ground TIV) shall be decoded as defined in table 5.51. |

Table 5.50: Vehicle target message (ground TIV) bit encoding

| Description | Octet | Bit number | | | | | | | |
|--|-------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| udid, ucd | 5 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| TIS-B message ID (tmi) TIV ID (tivid) | n | tid ₁ | tivid ₃ | tivid ₂ | tivid ₁ | tmi ₄ | tmi ₃ | tmi ₂ | tmi ₁ |
| target identifier (tid) | n + 1 | tid ₉ | tid ₈ | tid ₇ | tid ₆ | tid ₅ | tid ₄ | tid ₃ | tid ₂ |
| ADS-B fault flag (adsff) | n + 2 | tag ₅ | tag ₄ | tag ₃ | tag ₂ | tag ₁ | adsff | tid ₁₁ | tid ₁₀ |
| time stamp(tag) latitude (rlat) | n + 3 | rlat ₇ | rlat ₆ | rlat ₅ | rlat ₄ | rlat ₃ | rlat ₂ | rlat ₁ | tag ₆ |
| | n + 4 | rlat ₁₅ | rlat ₁₄ | rlat ₁₃ | rlat ₁₂ | rlat ₁₁ | rlat ₁₀ | rlat ₉ | rlat ₈ |
| longitude (rlg) | n + 5 | rlg ₃ | rlg ₂ | rlg ₁ | rlat ₂₀ | rlat ₁₉ | rlat ₁₈ | rlat ₁₇ | rlat ₁₆ |
| | n + 6 | rlg ₁₁ | rlg ₁₀ | rlg ₉ | rlg ₈ | rlg ₇ | rlg ₆ | rlg ₅ | rlg ₄ |
| | n + 7 | rlg ₁₉ | rlg ₁₈ | rlg ₁₇ | rlg ₁₆ | rlg ₁₅ | rlg ₁₄ | rlg ₁₃ | rlg ₁₂ |
| ground track (gtk) | n + 8 | gtk ₇ | gtk ₆ | gtk ₅ | gtk ₄ | gtk ₃ | gtk ₂ | gtk ₁ | rlg ₂₀ |
| ground speed (gsp) | n + 9 | gsp ₇ | gsp ₆ | gsp ₅ | gsp ₄ | gsp ₃ | gsp ₂ | gsp ₁ | gtk ₈ |

Table 5.51: Ground vehicle target message (ground TIV) field encoding

| Data field | Definition | Encoded value |
|-------------------|---|---------------|
| TIS-B message ID | See table 5.43 | 3 |
| TIV ID | Identifies TIV for ground stations configured to transmit more than one TIV | 0 to 7 |
| Target identifier | Unique 11-bit code allocated by ground systems to vehicle target | |
| ADS-B fault flag | ADS-B data corresponds with TIS-B data TIS-B data should be used in preference to ADS-B data | 0 1 |
| Time Stamp | See table 5.47 | |
| Latitude | Encoded as an offset from the reference position (see clause 5.2.8) | |
| Longitude | Encoded as an offset from the reference position (see clause 5.2.8) | |
| Ground speed | See ground speed encoding in table 5.14 | |
| Ground track | See ground track encoding in table 5.12 | |

5.2.8 TIS-B offset encoding

NOTE: The TIV vertex and target latitude and longitude positions are encoded as offsets from the reference position by subtracting the coordinates of the fixed reference position from those of the vertex or target position as defined in table 5.53. The TIV vertex and target longitude offsets from the reference position are multiplied by the function $\text{corr}(\text{lat})$ as defined in table 5.52, where $\text{lat} = \text{RND}(\text{latitude})$ is the vertex or target latitude rounded down to the nearest half degree.

Table 5.52: Values for scaling function $\text{corr}(\text{lat})$ used to encode target longitude

| lat | corr | lat | corr | lat | corr | lat | corr | lat | corr | lat | corr |
|------|-------|------|-------|------|-------|------|-------|------|-------|------|-------|
| 0 | 8 175 | 15 | 7 878 | 30 | 7 044 | 45 | 5 730 | 60 | 4 026 | 75 | 2 047 |
| 0,5 | 8 174 | 15,5 | 7 858 | 30,5 | 7 007 | 45,5 | 5 679 | 60,5 | 3 963 | 75,5 | 1 978 |
| 1 | 8 172 | 16 | 7 838 | 31 | 6 970 | 46 | 5 627 | 61 | 3 901 | 76 | 1 908 |
| 1,5 | 8 170 | 16,5 | 7 818 | 31,5 | 6 933 | 46,5 | 5 575 | 61,5 | 3 838 | 76,5 | 1 839 |
| 2 | 8 167 | 17 | 7 797 | 32 | 6 895 | 47 | 5 523 | 62 | 3 775 | 77 | 1 769 |
| 2,5 | 8 164 | 17,5 | 7 775 | 32,5 | 6 856 | 47,5 | 5 470 | 62,5 | 3 711 | 77,5 | 1 700 |
| 3 | 8 160 | 18 | 7 753 | 33 | 6 817 | 48 | 5 417 | 63 | 3 648 | 78 | 1 630 |
| 3,5 | 8 155 | 18,5 | 7 730 | 33,5 | 6 777 | 48,5 | 5 363 | 63,5 | 3 584 | 78,5 | 1 560 |
| 4 | 8 150 | 19 | 7 706 | 34 | 6 737 | 49 | 5 309 | 64 | 3 519 | 79 | 1 490 |
| 4,5 | 8 144 | 19,5 | 7 682 | 34,5 | 6 697 | 49,5 | 5 255 | 64,5 | 3 455 | 79,5 | 1 420 |
| 5 | 8 137 | 20 | 7 657 | 35 | 6 655 | 50 | 5 200 | 65 | 3 390 | 80 | 1 349 |
| 5,5 | 8 130 | 20,5 | 7 632 | 35,5 | 6 614 | 50,5 | 5 145 | 65,5 | 3 325 | 80,5 | 1 279 |
| 6 | 8 122 | 21 | 7 606 | 36 | 6 572 | 51 | 5 089 | 66 | 3 260 | 81 | 1 208 |
| 6,5 | 8 114 | 21,5 | 7 580 | 36,5 | 6 529 | 51,5 | 5 033 | 66,5 | 3 194 | 81,5 | 1 138 |
| 7 | 8 105 | 22 | 7 553 | 37 | 6 486 | 52 | 4 977 | 67 | 3 128 | 82 | 1 067 |
| 7,5 | 8 095 | 22,5 | 7 525 | 37,5 | 6 442 | 52,5 | 4 920 | 67,5 | 3 062 | 82,5 | 996 |
| 8 | 8 085 | 23 | 7 497 | 38 | 6 398 | 53 | 4 863 | 68 | 2 996 | 83 | 925 |
| 8,5 | 8 074 | 23,5 | 7 468 | 38,5 | 6 353 | 53,5 | 4 805 | 68,5 | 2 930 | 83,5 | 855 |
| 9 | 8 063 | 24 | 7 439 | 39 | 6 308 | 54 | 4 747 | 69 | 2 863 | 84 | 784 |
| 9,5 | 8 051 | 24,5 | 7 409 | 39,5 | 6 262 | 54,5 | 4 689 | 69,5 | 2 796 | 84,5 | 712 |
| 10 | 8 038 | 25 | 7 379 | 40 | 6 216 | 55 | 4 630 | 70 | 2 729 | 85 | 641 |
| 10,5 | 8 025 | 25,5 | 7 348 | 40,5 | 6 170 | 55,5 | 4 571 | 70,5 | 2 662 | 85,5 | 570 |
| 11 | 8 011 | 26 | 7 316 | 41 | 6 123 | 56 | 4 512 | 71 | 2 594 | 86 | 499 |
| 11,5 | 7 996 | 26,5 | 7 284 | 41,5 | 6 075 | 56,5 | 4 452 | 71,5 | 2 526 | 86,5 | 428 |
| 12 | 7 981 | 27 | 7 251 | 42 | 6 027 | 57 | 4 392 | 72 | 2 458 | 87 | 357 |
| 12,5 | 7 965 | 27,5 | 7 218 | 42,5 | 5 979 | 57,5 | 4 332 | 72,5 | 2 390 | 87,5 | 285 |
| 13 | 7 949 | 28 | 7 184 | 43 | 5 930 | 58 | 4 271 | 73 | 2 322 | 88 | 214 |
| 13,5 | 7 932 | 28,5 | 7 150 | 43,5 | 5 881 | 58,5 | 4 210 | 73,5 | 2 253 | 88,5 | 182 |
| 14 | 7 915 | 29 | 7 115 | 44 | 5 831 | 59 | 4 149 | 74 | 2 185 | 89 | 182 |
| 14,5 | 7 896 | 29,5 | 7 080 | 44,5 | 5 781 | 59,5 | 4 088 | 74,5 | 2 116 | 89,5 | 182 |

Table 5.53: Definition of offset encoding fields

| Data Field | Definition | Encoded Value |
|-----------------|--|---------------|
| Vertex latitude | Dlat = vertex latitude - reference latitude; Dlat is in degrees | |
| | For Dlat between -4 and $-(1+1/15)$ degrees: Encoded vertex latitude = $\text{Dlat} \times 15 + 60$ (Dlat has an increment of 1/15 of a degree or 4 NM; range from -240 NM to -64 NM) | 0 to 44 |
| | For Dlat between -1 and $-(1/3 + 1/60)$ degrees: Encoded vertex latitude = $\text{Dlat} \times 60 + 105$ (Dlat has an increment of 1/60 of a degree or 1 NM; range from -60 NM to -21 NM) | 45 to 84 |
| | For Dlat between -1/3 and +1/3 degrees: Encoded vertex latitude = $\text{Dlat} \times 120 + 125$ (Dlat has an increment of 1/120 of a degree or 0.5 NM; range from -20 NM to +20 NM) | 85 to 165 |
| | For Dlat between $(1/3 + 1/60)$ and +1 degrees: Encoded vertex latitude = $\text{Dlat} \times 60 + 145$ (Dlat has an increment of 1/60 of a degree or 1 NM; range from +21 NM to +60 NM) | 166 to 205 |
| | For Dlat between $(1+1/15)$ and +4 degrees: Encoded vertex latitude = $\text{Dlat} \times 15 + 190$ (Dlat has an increment of 1/15 of a degree or 4 NM; range from +64 NM to +240 NM) | 206 to 250 |
| | Unused | 251 to 255 |

| Data Field | Definition | Encoded Value |
|---------------------------|--|---|
| Vertex longitude | <p>Dlon = vertex longitude - reference longitude; Dlon is in degrees</p> <p>To encode a longitude offset in degrees, first convert to an offset in nautical miles (NM) (using the known latitude) in order to know which of the following encodings apply. Then perform the encoding with Dlon in degrees.</p> <p>For Dlon (converted to NM) between -240 NM and -64 NM: Encoded vertex longitude = $\text{INT}(\text{Dlon} \times \text{corr}(\text{vertex latitude}) / 545 + 60)$ (increment and range in degrees depends on latitude: increment in NM is 4 NM or 1/15 of a degree at equator; range is between -4 and $-(1+1/15)$ degrees at equator; range is between -180 and -48 degrees near the poles)</p> <p>For Dlon (converted to NM) between -60 NM and -21 NM: Encoded vertex longitude = $\text{INT}(\text{Dlon} \times \text{corr}(\text{vertex latitude}) \times 4 / 545 + 105)$ (increment and range in degrees depends on latitude: increment in NM is 1 NM or 1/60 of a degree at equator; range is between -1 and $-(1/3 + 1/60)$ degrees at equator; range is between -45 and -15,75 degrees near the poles)</p> <p>For Dlon (converted to NM) between -20 NM and +20 NM: Encoded vertex longitude = $\text{INT}(\text{Dlon} \times \text{corr}(\text{vertex latitude}) \times 8 / 545 + 125)$ (increment and range in degrees depends on latitude: increment in NM is 0,5 NM or 1/120 of a degree at equator; range is between -1/3 and +1/3 degrees at equator; range is between -15 and +15 degrees near the poles)</p> <p>For Dlon (converted to NM) between +21 NM and +60 NM: Encoded vertex longitude = $\text{INT}(\text{Dlon} \times \text{corr}(\text{vertex latitude}) \times 4 / 545 + 145)$ (increment and range in degrees depends on latitude: increment in NM is 1 NM or 1/60 of a degree at equator; range is between $(1/3 + 1/60)$ and +1 degrees at equator; range is between +15,75 and +45 degrees near the poles)</p> <p>For Dlon (converted to NM) between +64 NM and +240 NM: Encoded vertex longitude = $\text{INT}(\text{Dlon} \times \text{corr}(\text{vertex latitude}) / 545 + 190)$ (increment and range in degrees depends on latitude: increment in NM is 4 NM or 1/15 of a degree at equator; range is between $(1+1/15)$ and +4 degrees at equator; range is between +48 and +180 degrees near the poles)</p> <p>Unused</p> | <p>0 to 44</p> <p>45 to 84</p> <p>85 to 165</p> <p>166 to 205</p> <p>206 to 250</p> <p>251 to 255</p> |
| Airborne target latitude | <p>Dtlat = target latitude - reference latitude; Dtlat is in degrees</p> <p>For Dtlat between -4 and +4 degrees: Encoded target latitude = $\text{Dtlat} \times 8\,175 + 32\,700$ (Dtlat has an increment of 1/8 175 of a degree; target resolution ± 7 m; range from -240 NM to +240 NM)</p> <p>Unused</p> | <p>0 to 65 400</p> <p>65 401 to 65 536</p> |
| Airborne target longitude | <p>Dtlon = target longitude - reference longitude; Dtlon is in degrees</p> <p>For Dtlon (converted to NM) between -240 NM and +240 NM: Encoded target longitude = $\text{INT}(\text{Dtlon} \times \text{corr}(\text{target latitude}) + 32\,700)$ (Dtlon has an increment of 1/8 175 of a degree at the equator; target resolution ± 7 m; range in degrees depends on latitude: range is between -4 degrees and +4 degrees at the equator; range is between -180 degrees and +180 degrees near the poles)</p> <p>Unused</p> | <p>0 to 65 400</p> <p>65 401 to 65 536</p> |

| Data Field | Definition | Encoded Value |
|-------------------------|--|---------------------------|
| Ground target latitude | Dglat = target latitude - reference latitude; Dglat is in degrees For Dglat between -4 and +4 degrees: Encoded target latitude = $Dglat \times 130\,800 + 523\,200$ (Dglat has an increment of 1/130 800 of a degree; target resolution $\pm 0,45$ m; range from -240 NM to +240 NM) | 0 to 1 046 400 |
| | Unused | 1 046 401 to 1 048 575 |
| Ground target longitude | Dglon = target longitude - reference longitude; Dglon is in degrees For Dglon (converted to NM) between -240 NM and +240 NM: Encoded target longitude = $INT(Dglon \times \text{corr}(\text{target latitude}) \times 16 + 523\,200)$ (Dglon has an increment of 1/130 800 of a degree at the equator; target resolution $\pm 0,45$ m; range in degrees depends on latitude: range is between -4 degrees and +4 degrees at the equator; range is between -180 degrees and +180 degrees near the poles) | 0 to 1 046 400 |
| | Unused | 1 046 401 to 1 048 575 |

5.3 Requirements for FIS-B

5.3.1 Message format for received FIS-B messages

NOTE: The requirements in this clause do not apply to the FIS-B report request message.

| Requirement reference | |
|-----------------------|---|
| 5.3.1.1 | A FIS-B message shall be contained in a DLS UDATA DLPDU burst of type UINFO as described clause 5.3 of EN 302 842-2 [5]. |
| 5.3.1.2 | A station shall recognize a FIS-B service by the UDATA ID (udid) field being set to "00001" (see clause 5.3.1.3 of EN 302 842-2 [5]). |
| 5.3.1.3 | The FIS-B data shall be contained in the information field of a VDL Mode 4 burst, with the first message starting in bit 1 of octet 6. |
| 5.3.1.3a | Where more than one FIS-B message is contained in one FIS-B burst, each additional FIS-B message shall start in bit 1 of the octet following the last octet occupied by the previous FIS-B message. |
| 5.3.1.3b | Where more than one FIS-B message is contained in one FIS-B burst, if the previous FIS-B message does not finish at an octet boundary, up to 7 bits of zeros shall be included to complete the octet, prior to the start of the subsequent FIS-B message. |
| 5.3.1.4 | A station shall recognize each FIS-B message by the FIS-B message identifier, as defined in table 5.54. |
| 5.3.1.5 | To indicate a FIS-B message identifier greater than 14, bits 1 to 4 of octet 6 shall be set to 15, and a further 4 bits of fmi1 shall be inserted starting in bit 5 of octet 6 and ending in bit 8 of octet 6, so that $fmi = 15 + fmi1$. |

| Requirement reference | |
|-----------------------|--|
| 5.3.1.6 | <p>A station will be capable of decoding FIS-B bursts appended with the following reservation fields:</p> <ul style="list-style-type: none"> - a reservation ID (rid) equal to 1, with the burst appended by a null reservation field as defined in clause 5.2.9 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0 and an extended reservation ID (erid) equal to 00000binary, with the burst appended by a response reservation field as defined in clause 5.2.18 of EN 302 842-2 [5], with address type field equal to 7; - a reservation ID (rid) equal to 1, with the burst appended by a periodic broadcast reservation field as defined in clause 5.2.10 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0, with the burst appended by an incremental broadcast reservation field as defined in clause 5.2.11 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 1, with the burst appended by a combined periodic broadcast and incremental broadcast reservation field as defined in clause 5.2.12 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0, with the burst appended by a unicast reservation field with sdf=1 as defined in clause 5.2.14 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0, with the burst appended by a second frame reservation field as defined in clause 5.2.17 of EN 302 842-2 [5]; or - a reservation ID (rid) equal to 0, with the burst appended by a superframe reservation field as defined in clause 5.2.17 of EN 302 842-2 [5]. |

Table 5.54: FIS-B message identifier encoding

| Message | FIS-B Message ID Encoded value | 4-bit Message ID Extension (fmi1) Encoded value |
|---|--------------------------------|---|
| METAR | 0 | not present |
| ATIS | 1 | not present |
| RCN | 2 | not present |
| SIGMET | 3 | not present |
| SPECI | 4 | not present |
| REPORT REQUEST | 5 | not present |
| TSA | 6 | not present |
| Reserved for future use | 7 to 14 | not present |
| Available for future use with 4-bit extension | 15 | 0 to 15 |

5.3.2 Meteorological Aerodrome Report (METAR) message

| Requirement reference | |
|-----------------------|--|
| 5.3.2.1 | A station shall be capable of decoding a METAR message as defined in table 5.55. |
| 5.3.2.2 | Void. |
| 5.3.2.3 | Fields in the METAR report shall be decoded as defined in table 5.56. |
| 5.3.2.4 | If a TREND report is contained in the message, it shall always be received before the free text message. |
| 5.3.2.5 | If a TREND report is present, its end shall be recognized by the ASCII code "100100", corresponding to the character "\$". |
| 5.3.2.6 | Void. |
| 5.3.2.7 | ICAO Weather codes shall be decoded as defined in table 5.57. |
| 5.3.2.8 | Void. |
| 5.3.2.9 | When the CAVOK flag is set to "1", the fields for visibility (mas, masd, mis, misd), the 3 possible present weather fields (pw), the 4 possible cloud coverage fields (cc, ch, cty), the cloud number flag (cno), and the present weather flag (pwf) shall not be present in the received message. |

Table 5.55: METAR message bit encoding

| Description | Octet | Bit number | | | | | | | |
|---|-------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) FIS-B version ID (vers) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | fmi ₄ | fmi ₃ | fmi ₂ | fmi ₁ |
| message edition (med) local ID (locl) | 7 | locl ₅ | locl ₄ | locl ₃ | locl ₂ | locl ₁ | med ₃ | med ₂ | med ₁ |
| | 8 | locl ₁₃ | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ |
| date (date) | 9 | date ₁ | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ |
| time (ti) | 10 | ti ₄ | ti ₃ | ti ₂ | ti ₁ | date ₅ | date ₄ | date ₃ | date ₂ |
| wind direction (wdi) | 11 | wdi ₁ | ti ₁₁ | ti ₁₀ | ti ₉ | ti ₈ | ti ₇ | ti ₆ | ti ₅ |
| wind speed (wsp) | 12 | wsp ₃ | wsp ₂ | wsp ₁ | wdi ₆ | wdi ₅ | wdi ₄ | wdi ₃ | wdi ₂ |
| free text flag (ftxt) max wind flag (mxw) CAVOK flag (cav) TREND report flag (trd) | 13 | trd | cav | mxw | ftxt | wsp ₇ | wsp ₆ | wsp ₅ | wsp ₄ |
| present weather flag (pwf) recent weather flag (rwf) QFE flag (qfef) temperature (tem) | 14 | tem ₃ | tem ₂ | tem ₁ | qfef | rwf ₂ | rwf ₁ | pwf ₂ | pwf ₁ |
| dew point temperature (dew) | 15 | dew ₃ | dew ₂ | dew ₁ | tem ₈ | tem ₇ | tem ₆ | tem ₅ | tem ₄ |
| QNH (qnh) | 16 | qnh ₃ | qnh ₂ | qnh ₁ | dew ₈ | dew ₇ | dew ₆ | dew ₅ | dew ₄ |
| wind shear flag (wsf) sea flag (ssf) variable wind direction to (vdit) | 17 | vdit ₁ | ssf | wsf | qnh ₈ | qnh ₇ | qnh ₆ | qnh ₅ | qnh ₄ |
| variable wind direction from (vdif) | 18 | vdif ₃ | vdif ₂ | vdif ₁ | vdit ₆ | vdit ₅ | vdit ₄ | vdit ₃ | vdit ₂ |
| wind gust indicator (wgi) | 19 | wgi ₅ | wgi ₄ | wgi ₃ | wgi ₂ | wgi ₁ | vdif ₆ | vdif ₅ | vdif ₄ |
| minimum visibility (mis) | 20 | mis ₆ | mis ₅ | mis ₄ | mis ₃ | mis ₂ | mis ₁ | wgi ₇ | wgi ₆ |
| minimum visibility direction (misd) maximum visibility (mas) | 21 | mas ₅ | mas ₄ | mas ₃ | mas ₂ | mas ₁ | misd ₃ | misd ₂ | misd ₁ |
| maximum visibility direction (masd) present weather (pw) | 22 | pw ₄ | pw ₃ | pw ₂ | pw ₁ | masd ₃ | masd ₂ | masd ₁ | mas ₆ |
| cloud no. flag (cno) cloud coverage (cc) | 23 | cc ₃ | cc ₂ | cc ₁ | cno ₂ | cno ₁ | pw ₇ | pw ₆ | pw ₅ |
| cloud height (ch) | 24 | ch ₈ | ch ₇ | ch ₆ | ch ₅ | ch ₄ | ch ₃ | ch ₂ | ch ₁ |
| cloud type (cty) QFE (qfe) | 25 | qfe ₆ | qfe ₅ | qfe ₄ | qfe ₃ | qfe ₂ | qfe ₁ | cty ₂ | cty ₁ |
| recent weather (rw) | 26 | rw ₆ | rw ₅ | rw ₄ | rw ₃ | rw ₂ | rw ₁ | qfe ₈ | qfe ₇ |
| wind shear (ws) | 27 | ws ₇ | ws ₆ | ws ₅ | ws ₄ | ws ₃ | ws ₂ | ws ₁ | rw ₇ |
| sea-surface temperature (sst) state of the sea (ss) | 29 | ss ₁ | sst ₆ | sst ₅ | sst ₄ | sst ₃ | sst ₂ | sst ₁ | ws ₈ |
| free text message (txt) | 30 | txt ₅ | txt ₄ | txt ₃ | txt ₂ | txt ₁ | ss ₄ | ss ₃ | ss ₂ |
| | 31 | txt _j | txt _{j-1} | txt _{j-2} | | txt ₉ | txt ₈ | txt ₇ | txt ₆ |

..... [Denotes variable length field]

Table 5.56: METAR message field encoding

| Data field | Definition | Encoded value | Decoded value (if different) |
|--------------------|---|---|--|
| FIS-B Message ID | See table 5.54 | 0 | |
| Version ID | FIS-B as defined in the present document Reserved for future use | 0 1to15 | |
| Message edition | Edition of the METAR message (increments when message changes) A, B, C, D, E, F, G, H | 0 to 7 | |
| Local ID | ICAO airport designator 4 Letter Code, A to Z. Each letter encoded as a 5 bit binary A B C ↓ Y Z Unused | 00001 00010 00011 ↓ 11001 11010 11011 to 11111 | |
| Date | Unknown 1 2 3 ↓ 30 31 | 0 1 2 3 ↓ 30 31 | |
| Time (minutes) | Unknown $0 \leq \text{time} < 1$ $1 \leq \text{time} < 2$ $2 \leq \text{time} < 3$ $3 \leq \text{time} < 4$ ↓ $1\ 438 \leq \text{time} < 1\ 439$ $1\ 439 \leq \text{time} < 1\ 440$ Unused | 0 1 2 3 4 ↓ 1 439 1 440 1 441 to 2 047 | 0 1 2 3 1 438 1 439 |
| Wind Direction (°) | Unknown $355 \leq \text{heading} < 5$ $5 \leq \text{heading} < 15$ $15 \leq \text{heading} < 25$ $25 \leq \text{heading} < 35$ ↓ $315 \leq \text{heading} < 325$ $325 \leq \text{heading} < 335$ $335 \leq \text{heading} < 345$ $345 \leq \text{heading} < 355$ Unused | 0 1 2 3 4 ↓ 33 34 35 36 37 to 63 | 0 10 20 30 ↓ 320 330 340 350 |
| Wind Speed (kts) | Unknown $0 < \text{speed} < 0,5$ $0,5 \leq \text{speed} < 1,5$ $1,5 \leq \text{speed} < 2,5$ $2,5 \leq \text{speed} < 3,5$ ↓ $96,5 \leq \text{speed} < 97,5$ $97,5 \leq \text{speed} < 98,5$ $98,5 \leq \text{speed} < 99,5$ $99,5 \leq \text{speed} < 100,5$ $100,5 \leq \text{speed}$ Unused | 0 1 2 3 4 ↓ 98 99 100 101 102 103 to 127 | 0 1 2 3 ↓ 97 98 99 100 |

| Data field | Definition | Encoded value | Decoded value (if different) |
|-----------------------|--|--|--|
| Free text flag (ftxt) | Refers to following field; Free text in the free text field. This flag does not affect the existence of a TREND report in the free text field. None Message | 0 1 | |
| Max wind flag | Refers to following field; Wind Gust Indicator None Gusts present | 0 1 | |
| CAVOK flag | Refers to following field; Visibility (mas, masd, mis, misd), the 3 possible present weather fields (pw) and the 4 possible cloud coverage characteristics fields (cc, ch, cty), the cloud number flag (cno) and the present weather flag (pwf). Under CAVOK conditions these will not be sent. No CAVOK CAVOK | 0 1 | |
| TREND flag | No TREND report TREND report present | 0 1 | |
| Present weather flag | Number of present weather fields to follow (max 3) Refers to following field; Present weather 1 weather field included 2 weather fields included 3 weather fields included reserved for future use | 0 1 2 3 | |
| Recent weather flag | Number of recent weather fields to follow (max 3) Refers to following field; Recent weather 1 weather field included 2 weather fields included 3 weather fields included reserved for future use | 0 1 2 3 | |
| QFE flag | Present Not present | 0 1 | |
| Temperature (°C) | Unknown -80 < temp ≤ -79 -79 < temp ≤ -78 -78 < temp ≤ -77 ↓ -1 < temp ≤ 0 0 < temp ≤ 1 1 < temp ≤ 2 ↓ 59 < temp ≤ 60 60 < temp Unused | 0 1 2 3 ↓ 80 81 82 ↓ 140 141 142 to 255 | -79 -78 -77 ↓ 0 1 2 ↓ 60 |
| Dew point (°C) | As for temperature | | |
| QNH (hPa) | Unknown 850 ≤ QNH < 851 851 ≤ QNH < 852 852 ≤ QNH < 853 ↓ 1 097 ≤ QNH < 1 098 1 098 ≤ QNH < 1 099 1 099 ≤ QNH < 1 100 1 100 ≤ QNH < 1 101 Unused | 0 1 2 3 ↓ 248 249 250 251 252 to 255 | 850 851 852 ↓ 1 097 1 098 1 099 1 100 |

| Data field | Definition | Encoded value | Decoded value (if different) |
|--|--|---|---|
| Sea Flag | Refers to following fields: state of the sea (ss) and sea temperature (sst) Present Not present | 0 1 | |
| Variable wind direction to (degrees) | Unknown 355 ≤ variable < 5 5 ≤ variable < 15 15 ≤ variable < 25 25 ≤ variable < 35 ↓ 315 ≤ variable < 325 325 ≤ variable < 335 335 ≤ variable < 345 345 ≤ variable < 355 Unused | 0 1 2 3 4 ↓ 33 34 35 36 37 to 63 | 0 10 20 30 ↓ 320 330 340 350 |
| Variable wind direction from (degrees) | As above | | |
| Wind Gust Indicator | As for wind speed | | |
| Minimum visibility (metres) | Unknown 0 ≤ visibility < 100 100 ≤ visibility < 150 150 ≤ visibility < 200 200 ≤ visibility < 250 ↓ 700 ≤ visibility < 750 750 ≤ visibility < 800 800 ≤ visibility < 900 900 ≤ visibility < 1 000 ↓ 6 000 ≤ visibility < 7 000 7 000 ≤ visibility < 8 000 8 000 ≤ visibility < 9 000 9 000 ≤ visibility < 10 000 10 000 ≤ visibility < 11 000 Unused | 0 1 2 3 4 ↓ 14 15 16 17 ↓ 41 42 43 44 45 46 to 63 | 50 100 150 200 ↓ 700 750 800 900 ↓ 6 000 7 000 8 000 9 000 10 000 |
| Minimum visibility direction | North North East East South East South South West West North West | 0 1 2 3 4 5 6 7 | 0 1 2 3 4 5 6 7 |
| Maximum visibility (metres) | As for minimum visibility | | |
| Maximum visibility direction | As for minimum visibility direction | | |
| Present weather | See weather codes in table 5.57 Encoded values not specified in table 5.57 are unused | | |
| Cloud no. flag | Number of cloud fields to follow (max 4) Referred to following fields; Cloud coverage, Cloud Height, Cloud Type 1 2 3 4 | 0 1 2 3 | |

| Data field | Definition | Encoded value | Decoded value (if different) |
|---------------------------------|---|---|--|
| Cloud coverage | Unknown Few Scattered Broken Overcast Sky clear Unused | 0 1 2 3 4 5 6 to 7 | |
| Cloud height (m) | 0 ≤ heading < 60 60 ≤ heading < 90 90 ≤ heading < 120 120 ≤ heading < 150 150 ≤ heading < 180 ↓ 2 910 ≤ heading < 2 940 2 940 ≤ heading < 2 970 2 970 ≤ heading < 3 000 3 000 ≤ heading < 3 300 ↓ 19 800 ≤ heading < 20 100 20 100 ≤ heading < 20 400 Unused | 0 1 2 3 4 ↓ 96 97 98 99 ↓ 155 156 157 to 255 | 30 60 90 120 150 ↓ 2 910 2 940 2 970 3 000 ↓ 19 800 20 100 |
| Cloud type | No clouds of concern Cumulonimbus Towering cumulus Future use | 0 1 2 3 | |
| QFE (hPa) | Unknown 850 ≤ QFE < 851 851 ≤ QFE < 852 852 ≤ QFE < 853 ↓ 1 097 ≤ QFE < 1 098 1 098 ≤ QFE < 1 099 1 099 ≤ QFE < 1 100 1 100 ≤ QFE < 1 101 Unused | 0 1 2 3 ↓ 248 249 250 251 252 to 255 | 850 851 852 ↓ 1 097 1 098 1 099 1 100 |
| Recent weather | See weather codes in table 5.57 Encoded values not specified in table 5.57 are unused | | |
| Sea-surface temperature (°C) | Unknown -10 ≤ SST < -9 -9 ≤ SST < -8 -8 ≤ SST < -7 ↓ 36 ≤ SST < 37 37 ≤ SST < 38 38 ≤ SST < 39 39 ≤ SST < 40 40 ≤ SST Unused | 0 1 2 3 ↓ 47 48 49 50 51 52 to 63 | -10 -9 -8 ↓ 36 37 38 39 40 |

| Data field | Definition | Encoded value | Decoded value (if different) |
|-------------------|---|---------------|------------------------------|
| State of the sea | Unknown | 0 | |
| | 0 | 1 | |
| | 1 | 2 | |
| | 2 | 3 | |
| | 3 | 4 | |
| | 4 | 5 | |
| | 5 | 6 | |
| | 6 | 7 | |
| | 7 | 8 | |
| | 8 | 9 | |
| | 9 | 10 | |
| | Unused | 11 to 15 | |
| Free text message | Variable field length (6-bit converted ASCII characters) See table 5.57a | | |

NOTE: See WMO Publication No306, Manual on Codes Vol 1.1, Part A for interpretation.

Table 5.57: Encoding for ICAO Weather codes

| Explanation | Encoded Value |
|-------------------------------------|---------------|
| No precipitation during observation | |
| Fume - smoke | 4 |
| Dust haze | 5 |
| Rising dust and sand | 7 |
| Dust devil | 8 |
| Brune - mist | 10 |
| Mince fog - shallow fog | 11 |
| Mince fog | 12 |
| Thunderstorm | 17 |
| Funnel cloud | 18 |
| Recent drizzle | 20 |
| Recent rain | 21 |
| Recent snow | 22 |
| Recent rain and snow | 23 |
| Recent freezing rain | 24 |
| Recent showers | 25 |
| Recent snow showers | 26 |
| Recent grain(hail) | 27 |
| Recent thunderstorm | 29 |
| Sand or dust storm | 30 |
| Heavy sand storm | 33 |
| Low drifting snow | 36 |
| Blowing snow | 38 |
| Fog patches | 40 |
| Fog | 42 |
| Freezing fog | 48 |
| Precipitation during observation | |
| Drizzle | 50 |
| Heavy drizzle | 54 |
| Freezing drizzle | 56 |
| Heavy freezing drizzle | 57 |
| Rain | 58 |
| Heavy rain | 64 |
| Freezing rain | 66 |
| Heavy freezing rain | 67 |
| Rain and snow | 68 |
| Snow | 70 |
| Heavy snow | 74 |
| Snow grains | 77 |
| Ice pellets | 79 |
| Showers | 80 |
| Heavy showers | 81 |

| Explanation | Encoded Value |
|--------------------------|---------------|
| Showers of rain and snow | 83 |
| Snow showers | 85 |
| Soft hail | 87 |
| Hail | 89 |
| Thunderstorm | 95 |
| Thunderstorm with hail | 96 |
| Heavy thunderstorm | 97 |

Table 5.57a: Free text 6-bit character encoding (converted from ASCII 8-bit)

| First 32 characters | | | Last 32 characters | | |
|---------------------|-----------|----------------|--------------------|-----------|----------------|
| Character | (Decimal) | Encoded Binary | Character | (Decimal) | Encoded Binary |
| @ | 0 | 000000 | [space] | 32 | 100000 |
| A | 1 | 000001 | ! | 33 | 100001 |
| B | 2 | 000010 | " | 34 | 100010 |
| C | 3 | 000011 | # | 35 | 100011 |
| D | 4 | 000100 | \$ | 36 | 100100 |
| E | 5 | 000101 | % | 37 | 100101 |
| F | 6 | 000110 | & | 38 | 100110 |
| G | 7 | 000111 | ' | 39 | 100111 |
| H | 8 | 001000 | (| 40 | 101000 |
| I | 9 | 001001 |) | 41 | 101001 |
| J | 10 | 001010 | x | 42 | 101010 |
| K | 11 | 001011 | + | 43 | 101011 |
| L | 12 | 001100 | , | 44 | 101100 |
| M | 13 | 001101 | - | 45 | 101101 |
| N | 14 | 001110 | . | 46 | 101110 |
| O | 15 | 001111 | / | 47 | 101111 |
| P | 16 | 010000 | 0 | 48 | 110000 |
| Q | 17 | 010001 | 1 | 49 | 110001 |
| R | 18 | 010010 | 2 | 50 | 110010 |
| S | 19 | 010011 | 3 | 51 | 110011 |
| T | 20 | 010100 | 4 | 52 | 110100 |
| U | 21 | 010101 | 5 | 53 | 110101 |
| V | 22 | 010110 | 6 | 54 | 110110 |
| W | 23 | 010111 | 7 | 55 | 110111 |
| X | 24 | 011000 | 8 | 56 | 111000 |
| Y | 25 | 011001 | 9 | 57 | 111001 |
| Z | 26 | 011010 | : | 58 | 111010 |
| [| 27 | 011011 | ; | 59 | 111011 |
| \ | 28 | 011100 | < | 60 | 111100 |
|] | 29 | 011101 | = | 61 | 111101 |
| ^ | 30 | 011110 | > | 62 | 111110 |
| _ | 31 | 011111 | ? | 63 | 111111 |

5.3.3 Special Observations and Reports (SPECI) message

| Requirement reference | |
|-----------------------|---|
| 5.3.3.1 | A station will be capable of decoding a SPECI message, defined as for the METAR message in tables 5.55 and 5.56, with the FIS-B message ID set to 4, and the message edition indicating the edition of the SPECI message. |

5.3.4 Automatic Terminal Information Service (ATIS) message

| Requirement reference | |
|-----------------------|--|
| 5.3.4.1 | A station will be capable of decoding an ATIS message as defined in table 5.58. |
| 5.3.4.2 | If a TREND report is contained in the message, it shall always be received before the free text message. |
| 5.3.4.3 | If a TREND report is present, its end shall be recognized by the ASCII code "100100", corresponding to the character "\$". |
| 5.3.4.4 | Void. |
| 5.3.4.5 | Fields in the ATIS message shall be decoded as defined in table 5.59. |
| 5.3.4.6 | Void. |
| 5.3.4.7 | When the CAVOK flag is set to "1", the fields for visibility (mas, masd, mis, misd), the 3 possible present weather fields (pw), the 4 possible cloud coverage fields (cc, ch, cty), the cloud number flag (cno), and the present weather flag (pwf) shall not be present in the received message. |

Table 5.58: ATIS message bit encoding

| Description | Octet | Bit number | | | | | | | |
|--|-------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) FIS-B version ID (vers) message edition (med) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | fmi ₄ | fmi ₃ | fmi ₂ | fmi ₁ |
| local ID(locl) | 7 | locl ₅ | locl ₄ | locl ₃ | locl ₂ | locl ₁ | med ₃ | med ₂ | med ₁ |
| | 8 | locl ₁₃ | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ |
| date (date) | 9 | date ₁ | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ |
| time (ti) | 10 | ti ₄ | ti ₃ | ti ₂ | ti ₁ | date ₅ | date ₄ | date ₃ | date ₂ |
| holding delay (hd) | 11 | hd ₁ | ti ₁₁ | ti ₁₀ | ti ₉ | ti ₈ | ti ₇ | ti ₆ | ti ₅ |
| transition level (tran) | 12 | tran ₅ | tran ₄ | tran ₃ | tran ₂ | tran ₁ | hd ₄ | hd ₃ | hd ₂ |
| other op. activity flag (othf) other op. activity (oth) airport availability (aavl) | 13 | aavl ₃ | aavl ₂ | aavl ₁ | oth ₃ | oth ₂ | oth ₁ | othf | tran ₆ |
| | 14 | aavl ₁₁ | aavl ₁₀ | aavl ₉ | aavl ₈ | aavl ₇ | aavl ₆ | aavl ₅ | aavl ₄ |
| wind direction (wdi) wind speed (wsp) | 15 | wsp ₁ | wdi ₆ | wdi ₅ | wdi ₄ | wdi ₃ | wdi ₂ | wdi ₁ | aavl ₁₂ |
| temperature (tem) | 16 | tem ₂ | tem ₁ | wsp ₇ | wsp ₆ | wsp ₅ | wsp ₄ | wsp ₃ | wsp ₂ |
| dew point (dew) | 17 | dew ₂ | dew ₁ | tem ₈ | tem ₇ | tem ₆ | tem ₅ | tem ₄ | tem ₃ |
| QNH (qnh) | 18 | qnh ₂ | qnh ₁ | dew ₈ | dew ₇ | dew ₆ | dew ₅ | dew ₄ | dew ₃ |
| free text flag (ftxt) max wind flag (mxw) CAVOK flag (cav) | 19 | mxw | ftxt | qnh ₈ | qnh ₇ | qnh ₆ | qnh ₅ | qnh ₄ | qnh ₃ |
| TREND report flag (trd) QFE flag (qfef) wind shear flag (ws) Present weather flag (pwf) Recent weather flag (rwf) variable wind direction to (vdit) | 20 | rwf ₂ | rwf ₁ | pwf ₂ | pwf ₁ | ws | qfef | trd | cav |
| variable wind direction from (vdif) | 21 | vdif ₂ | vdif ₁ | vdit ₆ | vdit ₅ | vdit ₄ | vdit ₃ | vdit ₂ | vdit ₁ |
| wind gust indicator (wgi) | 22 | wgi ₄ | wgi ₃ | wgi ₂ | wgi ₁ | vdif ₆ | vdif ₅ | vdif ₄ | vdif ₃ |
| Minimum visibility (mis) | 23 | mis ₅ | mis ₄ | mis ₃ | mis ₂ | mis ₁ | wgi ₇ | wgi ₆ | wgi ₅ |
| Minimum visibility direction (misd) Maximum visibility (mas) | 24 | mas ₄ | mas ₃ | mas ₂ | mas ₁ | misd ₃ | misd ₂ | misd ₁ | mis ₆ |
| Maximum visibility direction (masd) cloud no. flag (cno) cloud coverage (cc) | 25 | cc ₁ | cno ₂ | cno ₁ | masd ₃ | masd ₂ | masd ₁ | mas ₆ | mas ₅ |
| cloud height (ch) | 26 | ch ₆ | ch ₅ | ch ₄ | ch ₃ | ch ₂ | ch ₁ | cc ₃ | cc ₂ |
| cloud type (cty) present weather (pw) | 27 | pw ₄ | pw ₃ | pw ₂ | pw ₁ | cty ₂ | cty ₁ | ch ₈ | ch ₇ |

| Description | Octet | Bit number | | | | | | | |
|-------------------------|-------|------------------|--------------------|--------------------|------------------|-------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| recent weather (rw) | 28 | rw ₅ | rw ₄ | rw ₃ | rw ₂ | rw ₁ | pw ₇ | pw ₆ | pw ₅ |
| QFE (qfe) | 29 | qfe ₆ | qfe ₅ | qfe ₄ | qfe ₃ | qfe ₂ | qfe ₁ | rw ₇ | rw ₆ |
| free text message (txt) | 30 | txt ₆ | txt ₅ | txt ₄ | txt ₃ | txt ₂ | txt ₁ | qfe ₈ | qfe ₇ |
| | 31 | txt _j | txt _{j-1} | txt _{j-2} | | txt ₁₀ | txt ₉ | txt ₈ | txt ₇ |

: : [Denotes variable length field]

Table 5.59: ATIS message field encoding

| Data field | Definition | Encoded value | Decoded value (if different) |
|---------------------------------|---|---------------|------------------------------|
| FIS-B Message ID | See table 5.54 | 1 | |
| Version ID | FIS-B as defined in the present document Reserved for future use | 0 1 to 15 | |
| Message edition | See definition in table 5.56 | | |
| Local ID | See definition in table 5.56 | | |
| Date | See definition in table 5.56 | | |
| Time | See definition in table 5.56 | | |
| Holding Delay | No Delay | 0 | |
| | 5 min | 1 | |
| | 10 min | 2 | |
| | 15 min | 3 | |
| | 20 min | 4 | |
| | 25 min | 5 | |
| | 30 min | 6 | |
| | 35 min | 7 | |
| | 40 min | 8 | |
| | 45 min | 9 | |
| | 50 min | 10 | |
| | 55 min | 11 | |
| | 60 min | 12 | |
| | Unused | 13 | |
| | Unused | 14 | |
| Unknown Delay | 15 | | |
| Transition level | Unknown | 0 | |
| | 0 | 1 | |
| | 5 | 2 | |
| | 10 | 3 | |
| | 15 | 4 | |
| | ↓ | ↓ | |
| | 285 | 58 | |
| | 290 | 59 | |
| | 295 | 60 | |
| | 300 | 61 | |
| | Reserved for future use | 62 to 63 | |
| Other operational activity flag | Present | 0 | |
| | Not present | 1 | |
| Other operational activity | Reserved for future use. (e.g. bird activity) | 0 to 7 | |
| Airport availability | Bits 1 to 11: as for Time in table 5.56 | | |
| | Bit 12: Open | 1 | |
| | Closed | 0 | |
| Wind Direction (°) | See definition in table 5.56 | | |
| Wind Speed (kts) | See definition in table 5.56 | | |
| Temperature (degrees) | See definition in table 5.56 | | |
| Dew point (degrees) | As for temperature | | |
| QNH | See definition in table 5.56 | | |

| Data field | Definition | Encoded value | Decoded value (if different) |
|---|--|---------------|------------------------------|
| Free text flag | Refers to following field: Free text in the free text field. This flag does not affect the existence of a TREND report in the free text field. None Message | 0 1 | |
| Max wind flag | Refers to following field: Wind Gust Indicator None Gusts present | 0 1 | |
| CAVOK flag | Refers to following fields: visibility (mas, masd, mis, misd), the 3 possible present weather fields (pw), the 4 possible cloud coverage fields (cc, ch, ct), the cloud number flag (cno), and the present weather flag (pwf). Under CAVOK conditions these will not be sent. No CAVOK CAVOK | 0 1 | |
| TREND flag | None Message | 0 1 | |
| QFE flag | Present Not present | 0 1 | |
| Present weather flag | See definition in table 5.56 | | |
| Recent weather flag | See definition in table 5.56 | | |
| Variable wind directions to (degrees) | See definition in table 5.56 | | |
| Variable wind directions from (degrees) | As above | | |
| Wind gust indicator | As for wind speed | | |
| Minimum visibility (metres) | See definition in table 5.56 | | |
| Minimum visibility direction | See definition in table 5.56 | | |
| Maximum visibility (metres) | As for minimum visibility | | |
| Maximum visibility direction | As for minimum visibility direction | | |
| Cloud no. flag | See definition in table 5.56 | | |
| Cloud coverage | See definition in table 5.56 | | |
| Cloud Height (m) | See definition in table 5.56 | | |
| Cloud Type | See definition in table 5.56 | | |
| Present weather | See weather codes in table 5.57 Encoded values not specified in table 5.57 are unused | | |
| Recent weather | See weather codes in table 5.57 Encoded values not specified in table 5.57 are unused | | |
| QFE | See definition in table 5.56 | | |
| Free text message | Variable field length (6-bit converted ASCII characters) See table 5.57a | | |

5.3.5 Runway Condition (RCN) message

| Requirement reference | |
|-----------------------|---|
| 5.3.5.1 | A station shall be capable of decoding an RCN message as defined in table 5.60. |
| 5.3.5.2 | Fields in the RCN message shall be decoded as defined in table 5.61. |

Table 5.60: RCN message bit encoding

| Description | Octet | Bit number | | | | | | | |
|---|-------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) FIS-B version ID (vers) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | fmi ₄ | fmi ₃ | fmi ₂ | fmi ₁ |
| message edition (med) local ID (locl) | 7 | locl ₅ | locl ₄ | locl ₃ | locl ₂ | locl ₁ | med ₃ | med ₂ | med ₁ |
| | 8 | locl ₁₃ | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ |
| date (date) | 9 | date ₁ | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ |
| time (ti) | 10 | ti ₄ | ti ₃ | ti ₂ | ti ₁ | date ₅ | date ₄ | date ₃ | date ₂ |
| braking action (brk) | 11 | brk ₁ | ti ₁₁ | ti ₁₀ | ti ₉ | ti ₈ | ti ₇ | ti ₆ | ti ₅ |
| runway ID (run) | 12 | run ₆ | run ₅ | run ₄ | run ₃ | run ₂ | run ₁ | brk ₃ | brk ₂ |
| Approach type flag (aptf) Approach type (apty) | 13 | apty ₄ | apty ₃ | apty ₂ | apty ₁ | aptf ₂ | aptf ₁ | run ₈ | run ₇ |
| reduced runway length (rrl) | 14 | rrl ₃ | rrl ₇ | rrl ₆ | rrl ₅ | rrl ₄ | rrl ₃ | rrl ₂ | rrl ₁ |
| reduced runway width (rrw) runway deposits (rdp) | 15 | rdp ₁ | rrw ₇ | rrw ₆ | rrw ₅ | rrw ₄ | rrw ₃ | rrw ₂ | rrw ₁ |
| runway contamination (rcon) depth of deposit (dod) | 16 | dod ₂ | dod ₁ | rcon ₃ | rcon ₂ | rcon ₁ | rdp ₄ | rdp ₃ | rdp ₂ |
| RVR touchdown (rvt) | 17 | rvt ₃ | rvt ₂ | rvt ₁ | dod ₇ | dod ₆ | dod ₅ | dod ₄ | dod ₃ |
| RVR midpoint (rvm) | 18 | rvm ₅ | rvm ₄ | rvm ₃ | rvm ₂ | rvm ₁ | rvt ₆ | rvt ₅ | rvt ₄ |
| RVR endpoint (rve) RVR source (rso) | 19 | rso ₁ | rve ₆ | rve ₅ | rve ₄ | rve ₃ | rve ₂ | rve ₁ | rvm ₆ |
| runway in use (rus) runway availability (ravl) | 20 | ravl ₅ | ravl ₄ | ravl ₃ | ravl ₂ | ravl ₁ | rus ₂ | rus ₁ | rso ₂ |
| wind shear | 21 | ws | ravl ₁₂ | ravl ₁₁ | ravl ₁₀ | ravl ₉ | ravl ₈ | ravl ₇ | ravl ₆ |

Table 5.61: RCN message field encoding

| Data field | Definition | Encoded Value | Decoded value (if different) |
|--------------------|---|--------------------------------------|------------------------------|
| FIS-B Message ID | See table 5.54 | 2 | |
| Version ID | FIS-B as defined in the present document Reserved for future use | 0 1 to 15 | |
| Message edition | See definition in table 5.56 | | |
| Local ID | See definition in table 5.56 | | |
| Date | See definition in table 5.56 | 0 | |
| Time | See definition in table 5.56 | 0 | |
| Braking action | Braking action not reported Braking action: Poor Braking action: Medium/Poor Braking action: Medium Braking action: Medium/Good Braking action: Good Figures unreliable Unused | 0 1 2 3 4 5 6 7 | |
| Runway ID | Bits 1 to 6: 01 to 36 Bits 7 to 8: Single runway Left runway Right runway Centre runway | 0 to 35 0 1 2 3 | |
| Approach type flag | 1 approach field included 2 approach field included 3 approach field included Unused | 0 1 2 3 | |

| Data field | Definition | Encoded Value | Decoded value (if different) |
|-----------------------------------|--------------------------------------|---------------|---------------------------------|
| Approach type | ILS CAT I | 0 | |
| | ILS CAT II | 1 | |
| | ILS CAT III | 2 | |
| | VOR | 3 | |
| | VOR/DME | 4 | |
| | NDB | 5 | |
| | Radar Vectoring | 6 | |
| | VDF | 7 | |
| | MLS | 8 | |
| | VISUAL | 9 | |
| Unused | 10 to 15 | | |
| Reduced runway length (metres) | Not reported | 0 | |
| | $0 \leq \text{length} < 50$ | 1 | 25 |
| | $50 \leq \text{length} < 100$ | 2 | 75 |
| | $100 \leq \text{length} < 150$ | 3 | 125 |
| | $150 \leq \text{length} < 200$ | 4 | 175 |
| | ↓ | ↓ | |
| | $6\ 300 \leq \text{length} < 6\ 350$ | 127 | 6 325 |
| | $6\ 350 \leq \text{length} < 6\ 400$ | 128 | 6 375 |
| | $6\ 400 \leq \text{length} < 6\ 450$ | 129 | 6 425 |
| | $6\ 450 \leq \text{length}$ | 130 | |
| | Unused | 131 to 254 | |
| | Full length | 255 | |
| Reduced runway width (metres) | Not reported | 0 | |
| | $0 \leq \text{width} < 1$ | 1 | 1 |
| | $1 \leq \text{width} < 2$ | 2 | 2 |
| | $2 \leq \text{width} < 3$ | 3 | 3 |
| | $3 \leq \text{width} < 4$ | 4 | 4 |
| | ↓ | ↓ | |
| | $67 \leq \text{width} < 68$ | 68 | 68 |
| | $68 \leq \text{width} < 69$ | 69 | 69 |
| | $69 \leq \text{width} < 70$ | 70 | 70 |
| | $70 \leq \text{width}$ | 71 | |
| | Unused | 72 to 126 | |
| | Full width | 127 | |
| Runway deposits | Clear and dry | 0 | |
| | Damp | 1 | |
| | Wet and water patches | 2 | |
| | Rime or frost covered | 3 | |
| | Dry snow | 4 | |
| | Wet snow | 5 | |
| | Slush | 6 | |
| | Ice | 7 | |
| | Compacted or rolled snow | 8 | |
| | Frozen ruts or ridges | 9 | |
| | Type of deposit not reported | 10 | |
| | Unused | 11 to 15 | |
| | Runway contamination | None | 0 |
| 10 % or less | | 1 | |
| 11 % to 25 % | | 2 | |
| 26 % to 50 % | | 3 | |
| 51 % to 100 % | | 4 | |
| Unused | | 5 | |
| Unused | | 6 | |
| Not reported | 7 | | |

| Data field | Definition | Encoded Value | Decoded value (if different) |
|---|---|---------------|---------------------------------|
| Depth of Deposit | less than 1 mm | 0 | |
| | 1 mm | 1 | |
| | 2 mm | 2 | |
| | 3 mm | 3 | |
| | ↓ | ↓ | |
| | 89 mm | 89 | |
| | 90 mm | 90 | |
| | 10 cm | 91 | |
| | 15 cm | 92 | |
| | 20 cm | 93 | |
| | ↓ | ↓ | |
| | 35 cm | 96 | |
| | 40 cm or more | 97 | |
| | Runway not operational due to snow, slush, ice, large drifts or runway clearance, and depth not reported. | 98 | |
| Depth deposit operationally not significant or not measurable. | 99 | | |
| Unused | 100 to 127 | | |
| RVR - Touchdown (metres) | Not reported | 0 | |
| | 0 < RVR < 25 | 1 | 0 |
| | 25 < RVR < 50 | 2 | 25 |
| | 50 < RVR < 75 | 3 | 50 |
| | ↓ | ↓ | |
| | 350 < RVR < 375 | 15 | 350 |
| | 375 < RVR < 400 | 16 | 375 |
| | 400 < RVR < 450 | 17 | 400 |
| | 450 < RVR < 500 | 18 | 450 |
| | ↓ | ↓ | |
| | 700 < RVR < 750 | 23 | 700 |
| | 750 < RVR < 800 | 24 | 750 |
| | 800 < RVR < 900 | 25 | 800 |
| | 900 < RVR < 1 000 | 26 | 900 |
| | ↓ | ↓ | |
| 1 300 < RVR < 1 400 | 30 | 1 300 | |
| 1 400 < RVR < 1 500 | 31 | 1 400 | |
| 1 500 < RVR < 1 600 | 32 | 1 500 | |
| Unused | 33 to 63 | | |
| RVR - Midpoint | As for RVR - Touchdown | | |
| RVR - Endpoint | As for RVR - Touchdown | | |
| RVR source | Unknown | 0 | |
| | Human Observer | 1 | |
| | Instrumented RVR system (IRVR) | 2 | |
| | Reserved for future use | 3 | |
| Runway in use | Yes - for arrivals | 0 | |
| | Yes - for departures | 1 | |
| | Yes - mixed operations | 2 | |
| | No | 3 | |
| Runway availability | Bits 1 to 11: as for Time in table 5.56 | | |
| | Bit 12: Open | 1 | |
| | Closed | 0 | |
| Windshear | No | 0 | |
| | Yes | 1 | |

Table 5.63: SIGMET message field encoding

| Data field | Definition | Encoded value |
|---------------------------------|---|---------------|
| FIS-B Message ID | See table 5.54 | 3 |
| FIS-B sequence number | 1 to 16 | 0 to 15 |
| Message edition | See definition in table 5.56 | |
| Location indicator | See definition for local ID in table 5.56 | |
| Originator indicator | WMO originator office encoded as ICAO aerodrome | |
| Date issued | See definition in table 5.56 | |
| Time issued | See definition in table 5.56 | |
| Date valid to | See definition in table 5.56 | |
| Time valid to | See definition in table 5.56 | |
| Free text flag | None | 0 |
| | Message | 1 |
| SIGMET flag | Normal | 0 |
| | Convective SIGMET | 1 |
| Phenomenon / Condition reported | Condition reported in free text | 0 |
| | Thunderstorm | 1 |
| | Thunderstorm obscured | 2 |
| | Thunderstorm embedded | 3 |
| | Thunderstorm frequent | 4 |
| | Thunderstorm squall line | 5 |
| | Thunderstorm obscured with heavy hail | 6 |
| | Thunderstorm embedded with heavy hail | 7 |
| | Thunderstorm frequent with heavy hail | 8 |
| | Thunderstorm squall line with heavy hail | 9 |
| | Tropical cyclone | 10 |
| | Freezing rain | 11 |
| | Moderate turbulence | 12 |
| | Severe turbulence | 13 |
| | Severe icing | 14 |
| | Severe icing due to freezing rain | 15 |
| | Severe mountain waves | 16 |
| Heavy dust storm | 17 | |
| Heavy sand storm | 18 | |
| Volcanic ash | 19 | |
| Isolated cumulonimbus | 20 | |
| Occasional cumulonimbus | 21 | |
| Frequent cumulonimbus | 22 | |
| Reserved for future use | 23 to 31 | |
| Free text message | Variable field length (6-bit converted ASCII characters) See table 5.57a | |

5.3.7 Temporary Segregated Areas (TSA) message

| Requirement reference | |
|-----------------------|---|
| 5.3.7.1 | A station shall be capable of decoding a TSA message as defined in table 5.64 and either table 5.65 or table 5.66. |
| 5.3.7.2 | The fixed field defined in table 5.64 shall be present in all TSA messages. |
| 5.3.7.3 | If the message is of type Control Zone, the octets defined in table 5.65 shall be appended to the fixed field at octet 10. |
| 5.3.7.4 | If the message is of type Restricted Area, Danger Area or Prohibited Area, the octets defined in table 5.66 shall be appended to the fixed field at octet 10. |
| 5.3.7.5 | Fields within the TSA message shall be decoded as defined in table 5.67. |

Table 5.64: TSA message fixed field bit encoding

| Description | Octet | Bit number | | | | | | | |
|--|-------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) version ID (vers) message edition (med) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | fmi ₄ | fmi ₃ | fmi ₂ | fmi ₁ |
| date (date) | 7 | date ₅ | date ₄ | date ₃ | date ₂ | date ₁ | med ₃ | med ₂ | med ₁ |
| time (ti) | 8 | ti ₈ | ti ₇ | ti ₆ | ti ₅ | ti ₄ | ti ₃ | ti ₂ | ti ₁ |
| Number of TSAs (NoT) | 9 | NoT ₅ | NoT ₄ | NoT ₃ | NoT ₂ | NoT ₁ | ti ₁₁ | ti ₁₀ | ti ₉ |
| Information field | 10-n | | | | | | | | |

Table 5.65: TSA message bit encoding for information field of type Control Zone

| Description | Octet | Bit number | | | | | | | |
|--|-------|---------------------|---------------------|---------------------|---------------------|---------------------|--------------------|---------------------|---------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| Type of Area (toa) Active flag (act) local ID (locl) | 10 | locl ₄ | locl ₃ | locl ₂ | locl ₁ | act ₁ | toa ₃ | toa ₂ | toa ₁ |
| | 11 | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ | locl ₅ |
| | 12 | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ | locl ₁₃ |
| supplement to local ID (sloc) active from/to date (actdat) active from/to time (actim) | 13 | actdat ₃ | actdat ₂ | actdat ₁ | sloc ₅ | sloc ₄ | sloc ₃ | sloc ₂ | sloc ₁ |
| | 14 | actim ₆ | actim ₅ | actim ₄ | actim ₃ | actim ₂ | actim ₁ | actdat ₅ | actdat ₄ |
| TWR frequency (freq) | 15 | freq ₃ | freq ₂ | freq ₁ | actim ₁₁ | actim ₁₀ | actim ₉ | actim ₈ | actim ₇ |
| | 16 | freq ₁₁ | freq ₁₀ | freq ₉ | freq ₈ | freq ₇ | freq ₆ | freq ₅ | freq ₄ |
| next TSA..... | 17-n | | | | | | | | |

Table 5.66: TSA message bit encoding for Restricted, Danger, or Prohibited area

| Description | Octet | Bit number | | | | | | | |
|---|-------|---------------------|---------------------|--------------------|---------------------|---------------------|---------------------|---------------------|---------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| Type of Area (toa) Active flag (act) state ID (stid) | 10 | stid ₄ | stid ₃ | stid ₂ | stid ₁ | act ₁ | toa ₃ | toa ₂ | toa ₁ |
| Area leap number (aln) | 11 | aln ₂ | aln ₁ | stid ₁₀ | stid ₉ | stid ₈ | stid ₇ | stid ₆ | stid ₅ |
| | 12 | aln ₁₀ | aln ₉ | aln ₈ | aln ₇ | aln ₆ | aln ₅ | aln ₄ | aln ₃ |
| Supplement to area leap number (saln) | 13 | saln ₅ | saln ₄ | saln ₃ | saln ₂ | saln ₁ | aln ₁₃ | aln ₁₂ | aln ₁₁ |
| active from/to date (actdat) active from/to time (actim) | 14 | actim ₃ | actim ₂ | actim ₁ | actdat ₅ | actdat ₄ | actdat ₃ | actdat ₂ | actdat ₁ |
| | 15 | actim ₁₁ | actim ₁₀ | actim ₉ | actim ₈ | actim ₇ | actim ₆ | actim ₅ | actim ₄ |
| next TSA..... | 16-n | | | | | | | | |

Table 5.67: TSA message field encoding

| Data field | Definition | Encoded Value |
|------------------------|--|--|
| FIS-B Message ID | See table 5.54 | 6 |
| Version ID | FIS-B as defined in the present document Reserved for future use | 0 1 to 15 |
| Message edition | See definition in table 5.56 | |
| Local ID | See definition in table 5.56 | |
| Date | See definition in table 5.56 | |
| Time | See definition in table 5.56 | |
| Number of TSA:s | 1 TSA 2 TSAs ↓ 32 TSAs | 0 1 ↓ 31 |
| Type of Area | Describes the type of TSA: Control zone Restricted Area Danger Area Prohibited Area Reserved for future use | 0 1 2 3 4 to 7 |
| ACTIVE FLAG | Denotes whether the TSA is active or not (A TSA could be transmitted a certain time before it becomes active) Not Active Active | 0 1 |
| Supplement to local ID | Some control zones can have different extensions depending on neighbouring control zones being open or not. To distinguish between the different parts of the control zone referred to, a letter is added to the local ID, this letter correspond to a certain part of the control zone. 1 Letter code A-Z encoded as a 5-bit binary. A B C ↓ Y Z | 00001 00010 00011 ↓ 11001 11010 |
| Active from/to date | If the TSA is not yet active (active flag set to 0) this field shows which date the area will become active. If the TSA is already active (active flag set to 1) this field shows which date the area will become not active. See date definition in table 5.56 | |
| Active from/to time | If the TSA is not yet active (active flag set to 0) this field shows which time the area will become active. If the TSA is already active (active flag set to 1) this field shows which time the area will become not active. See time definition in table 5.56 | |
| TWR frequency | The frequency used to contact TWR. Unknown 108,000 108,025 108,050 ↓ 136,925 136,950 136,975 Unused | 0 1 2 3 ↓ 1 158 1 159 1 160 1 161 to 2 047 |
| State ID | Each restricted, danger or prohibited area has a state ID connected to it, for example ES for Sweden. 2 letter code. Each letter encoded as a 5 bit binary. A B C ↓ Y Z | 00001 00010 00011 ↓ 11001 11010 |

| Data field | Definition | Encoded Value |
|--------------------------------|--|--|
| Area leap number | All restricted, danger and prohibited areas have a leap number attached to it, for example Restricted area number 242 in Sweden is written as ES R242. Area leap number in this case would be 242. 1 2 ↓ 8192 | 0 1 ↓ 8191 |
| Supplement to area leap number | A restricted, danger and prohibited area can be divided into several smaller areas denoted with a letter after the leap number. For example ES R242C means sub area C of restricted area 242 in Sweden. Thus, the supplement to the area number would be C in this case. 1 letter encoded as a 5 bit binary. A B C ↓ Y Z | 00001 00010 00011 ↓ 11001 11010 |

5.3.8 FIS-B report request message

| Requirement reference | |
|-----------------------|---|
| 5.3.8.1 | The FIS-B report request message shall use the general request burst format as defined in EN 302 842-2 [5], table 5.44. |
| 5.3.8.2 | The FIS-B report request message shall be encoded as defined in table 5.68. |
| 5.3.8.3 | Fields within the FIS-B report request message shall be encoded as defined in table 5.69. |
| 5.3.8.4 | The requested message ID (r-mi) shall be set to the FIS-B service indicator "00001111". |
| 5.3.8.5 | The FIS-B message identifier (fmi) shall indicate a report request message and be encoded as defined in table 5.54. |
| 5.3.8.6 | The message request field (mrq) shall be used to request that a particular FIS-B message is transmitted by the addressed station and be encoded as defined in table 5.54. |
| 5.3.8.7 | A station transmitting a report request message shall append to the message one of the following VDL Mode 4 reservation types: <ul style="list-style-type: none"> - a null reservation; - a response burst with a specific ground station address, or broadcast using address type field = 7; - a unicast reservation with a specific ground station address, or broadcast using address type field = 7; - a periodic broadcast; - an incremental broadcast; - a combined periodic and incremental broadcast; - a BND reservation. |
| 5.3.8.8 | Void. |
| 5.3.8.9 | Void. |
| 5.3.8.10 | Void. |
| 5.3.8.11 | Void. |

Table 5.68: FIS-B report request message bit encoding

| Description | Octet | Bit number | | | | | | | |
|--|-------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| mi = general request burst r-mi = FIS-B service indicator | 5 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| FIS-B message ID (fmi) message requested (mrq) | 6 | mrq ₁ | fmi ₄ | fmi ₃ | fmi ₂ | fmi ₁ | 0 | 0 | 0 |
| local ID (locl) | 7 | locl ₅ | locl ₄ | locl ₃ | locl ₂ | locl ₁ | mrq ₄ | mrq ₃ | mrq ₂ |
| | 8 | locl ₁₃ | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ |
| reserved (res) | 9 | res | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ |

Table 5.69: FIS-B report request message field encoding

| Data field | Definition | Encoded value |
|-------------------------|-------------------------------|---------------|
| FIS-B Message ID | See table 5.54. | 5 |
| message requested (mrq) | See table 5.54. | |
| Local ID | See definition in table 5.56. | |

5.3.9 Void

5.3.9a Airborne system functions

5.3.9a.1 Transmission of requests

| Requirement reference | |
|-----------------------|---|
| 5.3.9a.1.1 | The airborne system shall have the ability to transmit requests for information using the report request message. |

5.3.9a.2 Differentiation between reports

| Requirement reference | |
|-----------------------|---|
| 5.3.9a.2.1 | Airborne stations shall differentiate between FIS-B reports using: a) the most recent message edition number b) where the message edition numbers are the same, the most recently received message. |

5.4 Requirements for GNSS Augmentation Service Broadcast (GNS-B) message

5.4.1 Message format

| Requirement reference | |
|-----------------------|--|
| 5.4.1.1 | A GNS-B message shall be contained in a DLS UDATA DLPDU burst of type UINFO as described in clause 5.3 of EN 302 842-2 [5]. |
| 5.4.1.2 | A station shall recognize a GNS-B service by the UDATA ID (udid) field being set to "00011" (see clause 5.3.1.3 of EN 302 842-2 [5]). |
| 5.4.1.3 | GNS-B messages shall be contained in the information field of a VDL Mode 4 burst, with the first message starting in bit 1 of octet 6. |
| 5.4.1.3a | Where more than one GNS-B message is contained in one GNS-B burst, each additional GNS-B message shall start in bit 1 of the octet following the last octet occupied by the previous GNS-B message. |
| 5.4.1.3b | Where more than one GNS-B message is contained in one GNS-B burst, if the previous GNS-B message does not finish at an octet boundary, up to 7 bits of zeros shall be included to complete the octet, prior to the start of the subsequent GNS-B message. |
| 5.4.1.4 | A station shall recognize each GNS-B message by the GNS-B message identifier, as defined in table 5.70. |
| 5.4.1.5 | A station shall be capable of decoding GNS-B bursts appended with the following reservation fields: <ul style="list-style-type: none"> - a reservation ID (rid) equal to 1, with the burst appended by a null reservation field as defined in clause 5.9 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0 and an extended reservation ID (erid) equal to 00000binary, with the burst appended by a response reservation field as defined in clause 5.2.18 of EN 302 842-2 [5], with address type field equal to 7; - a reservation ID (rid) equal to 1, with the burst appended by a periodic broadcast reservation field as defined in clause 5.2.10 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0, with the burst appended by an incremental broadcast reservation field as defined in clause 5.2.11 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 1, with the burst appended by a combined periodic broadcast and incremental broadcast reservation field as defined in clause 5.2.12 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0, with the burst appended by a unicast reservation field with sdf=1 as defined in clause 5.2.14 of EN 302 842-2 [5]; - a reservation ID (rid) equal to 0, with the burst appended by a second frame reservation field as defined in clause 5.2.17 of EN 302 842-2 [5]; or - a reservation ID (rid) equal to 0, with the burst appended by a superframe reservation field as defined in clause 5.2.17 of EN 302 842-2 [5]. |

Table 5.70: GNS-B message identifier (gmi)

| Message | GNS-B Message ID Encoded value |
|--|--------------------------------|
| Type 1 Message (Pseudorange corrections) | 1 |
| Type 2 Message (GNS-B related data) | 2 |
| Type 4 Message (Final Approach Segment Data) | 4 |

5.4.2 Message Type 1

| Requirement reference | |
|-----------------------|---|
| 5.4.2.1 | A station shall be capable of decoding a GNS-B message type 1 as defined in table 5.71. |
| 5.4.2.2 | Fields in the message type 1 shall be decoded as defined in table 5.72. |
| 5.4.2.3 | Data for up to 12 satellites shall be accepted in Message Type 1. |
| 5.4.2.4 | The GPS Ephemeris decorrelation parameter and Ephemeris CRC may both be set to zero in Message Type 1 if the GPS receivers do not provide GPS Ephemeris, in which case they shall be ignored. |
| 5.4.2.5 | The GNS-B message CRC shall be decoded in accordance with the algorithm defined in clause 5.4.5.1. |
| 5.4.2.6 | All of B1, B2, B3 and B4 may be set to Bin10000000 in one or more measurement blocks, in which case they shall be ignored for that measurement block. |

Table 5.71: GNS-B Message Type 1 bit encoding

| Description | Octet | Bit number | | | | | | | |
|--|-------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| | 5 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| message block identifier (mbi) | 6 | mbi ₈ | mbi ₇ | mbi ₆ | mbi ₅ | mbi ₄ | mbi ₃ | mbi ₂ | mbi ₁ |
| GNS-B ID (gid) | 7 | gid ₈ | gid ₇ | gid ₆ | gid ₅ | gid ₄ | gid ₃ | gid ₂ | gid ₁ |
| | 8 | gid ₁₆ | gid ₁₅ | gid ₁₄ | gid ₁₃ | gid ₁₂ | gid ₁₁ | gid ₁₀ | gid ₉ |
| | 9 | gid ₂₄ | gid ₂₃ | gid ₂₂ | gid ₂₁ | gid ₂₀ | gid ₁₉ | gid ₁₈ | gid ₁₇ |
| message identifier (gmi) | 10 | 0 | 0 | 0 | 0 | 0 | gmi ₃ | gmi ₂ | gmi ₁ |
| message length (len) | 11 | len ₈ | len ₇ | len ₆ | len ₅ | len ₄ | len ₃ | len ₂ | len ₁ |
| modified Z-count (zc) | 12 | zc ₈ | zc ₇ | zc ₆ | zc ₅ | zc ₄ | zc ₃ | zc ₂ | zc ₁ |
| additional message flag (amf) | 13 | amf ₂ | amf ₁ | zc ₁₄ | zc ₁₃ | zc ₁₂ | zc ₁₁ | zc ₁₀ | zc ₉ |
| number of measurements (n) | 14 | t ₃ | t ₂ | t ₁ | n ₅ | n ₄ | n ₃ | n ₂ | n ₁ |
| measurement type (t) | | | | | | | | | |
| ephemeris decorrelation parameter (edp) | 15 | edp ₈ | edp ₇ | edp ₆ | edp ₅ | edp ₄ | edp ₃ | edp ₂ | edp ₁ |
| ephemeris CRC(ec) | 16 | ec ₈ | ec ₇ | ec ₆ | ec ₅ | ec ₄ | ec ₃ | ec ₂ | ec ₁ |
| | 17 | ec ₁₆ | ec ₁₅ | ec ₁₄ | ec ₁₃ | ec ₁₂ | ec ₁₁ | ec ₁₀ | ec ₉ |
| source availability duration (ad) | 18 | ad ₈ | ad ₇ | ad ₆ | ad ₅ | ad ₄ | ad ₃ | ad ₂ | ad ₁ |
| For N measurement blocks (i=1 to i=N): ranging source ID (id) | 19+(i-1)x11 | id ₈ | id ₇ | id ₆ | id ₅ | id ₄ | id ₃ | id ₂ | id ₁ |
| issue of data (iod) | 20+(i-1)x11 | iod ₈ | iod ₇ | iod ₆ | iod ₅ | iod ₄ | iod ₃ | iod ₂ | iod ₁ |
| pseudorange correction (prc) | 21+(i-1)x11 | prc ₈ | prc ₇ | prc ₆ | prc ₅ | prc ₄ | prc ₃ | prc ₂ | prc ₁ |
| | 22+(i-1)x11 | prc ₁₆ | prc ₁₅ | prc ₁₄ | prc ₁₃ | prc ₁₂ | prc ₁₁ | prc ₁₀ | prc ₉ |
| range rate correction (rrc) | 23+(i-1)x11 | rrc ₈ | rrc ₇ | rrc ₆ | rrc ₅ | rrc ₄ | rrc ₃ | rrc ₂ | rrc ₁ |
| | 24+(i-1)x11 | rrc ₁₆ | rrc ₁₅ | rrc ₁₄ | rrc ₁₃ | rrc ₁₂ | rrc ₁₁ | rrc ₁₀ | rrc ₉ |
| σ_{pr_gnd} (sd) | 25+(i-1)x11 | sd ₈ | sd ₇ | sd ₆ | sd ₅ | sd ₄ | sd ₃ | sd ₂ | sd ₁ |
| integrity parameter B1 (b1) | 26+(i-1)x11 | b1 ₈ | b1 ₇ | b1 ₆ | b1 ₅ | b1 ₄ | b1 ₃ | b1 ₂ | b1 ₁ |
| integrity parameter B2 (b2) | 27+(i-1)x11 | b2 ₈ | b2 ₇ | b2 ₆ | b2 ₅ | b2 ₄ | b2 ₃ | b2 ₂ | b2 ₁ |
| integrity parameter B3 (b3) | 28+(i-1)x11 | b3 ₈ | b3 ₇ | b3 ₆ | b3 ₅ | b3 ₄ | b3 ₃ | b3 ₂ | b3 ₁ |
| integrity parameter B4 (b4) | 29+(i-1)x11 | b4 ₈ | b4 ₇ | b4 ₆ | b4 ₅ | b4 ₄ | b4 ₃ | b4 ₂ | b4 ₁ |
| GNS-B message CRC (gc) | 30+(N-1)x11 | gc ₈ | gc ₇ | gc ₆ | gc ₅ | gc ₄ | gc ₃ | gc ₂ | gc ₁ |
| | 31+(N-1)x11 | gc ₁₆ | gc ₁₅ | gc ₁₄ | gc ₁₃ | gc ₁₂ | gc ₁₁ | gc ₁₀ | gc ₉ |
| | 32+(N-1)x11 | gc ₂₄ | gc ₂₃ | gc ₂₂ | gc ₂₁ | gc ₂₀ | gc ₁₉ | gc ₁₈ | gc ₁₇ |
| | 33+(N-1)x11 | gc ₃₂ | gc ₃₁ | gc ₃₀ | gc ₂₉ | gc ₂₈ | gc ₂₇ | gc ₂₆ | gc ₂₅ |

Table 5.72: GNS-B Message Type 1 field encoding

| Data field | Definition | Encoded Value |
|--------------------------------------|---|---|
| Message Identifier (gmi) | See table 5.70 | 1 |
| GNS-B message block identifier (mbi) | This represents the operating mode of the GNS-B message block. Normal GNS-B message Test GNS-B message | 10101010 11111111 |
| GNS-B ID (gid) | 4-character GNS-B identification included to differentiate between broadcasting stations. Each character is coded using the lower 6 bits of its International Alphabet No. 5 (IA-5) representation. Only upper case letters, numeric digits and IA-5 "blank" are used. The right-most character is transmitted first. For a 3-character GNS-B ID, the right most (first transmitted) character shall be IA-5 "blank". NOTE: The GNS-B ID is normally identical to the location indicator at the nearest airport. Assignment of GNS-B IDs will be co-ordinated as appropriate to avoid conflicts. | |
| Message length (len) | length of the message in 8-bit bytes including the 6-byte message block header (i.e. GNS-B Message block identifier, GNS-B ID, Message Type Identifier, Message Length), the message and the 4-byte message CRC code. Unused 10 bytes 11 bytes 255 bytes | 0 to 9 10 11 255 |
| Modified Z-count (zc) | The modified Z-count defines the reference time for all the message parameters in this message (including pseudorange correction and range-rate correction). The modified Z-count resets on the hour (xx:00), 20 minutes past the hour (xx:20) and 40 minutes past the hour (xx:40) referenced to GPS time. 00 minutes 00,0 s 00 minutes 00,1 s 00 minutes 00,2 s 01 minute 00,0 s 01 minute 00,1 s 19 minutes 59,9 s 20 minutes 00,0 s | 0 1 2 600 601 11999 12000 |
| Additional message flag (amf) | Identifies whether measurement blocks are contained in one or two Type 1 messages in a single frame. All measurement blocks are contained in one Type 1 Message This is the first of two Type 1 Messages in a frame containing measurement blocks Unused This is the second of two Type 1 Messages in a frame containing measurement blocks | 0 1 2 3 |
| Number of measurements (n) | This parameter identifies the number of measurement blocks in the message. 0 blocks 1 block 12 blocks | 0 1 12 |

| Data field | Definition | Encoded Value |
|---|---|---|
| Measurement type (t) | This parameter identifies the type of ranging signal from which the corrections have been computed C/A or CSA code L1 Reserved Reserved Reserved Unused | 0 1 2 3 4 to 7 |
| Ephemeris decorrelation parameter (edp) | This parameter characterizes the impact of residual ephemeris errors due to decorrelation for the first measurement block in the message SBAS Geostationary Satellite 0 m/m 5×10^{-6} m/m 10×10^{-6} m/m $1,275 \times 10^{-3}$ m/m | 0 0 1 2 255 |
| Ephemeris CRC (ec): | This parameter characterizes the impact of residual ephemeris errors due to decorrelation for the first measurement block in the message | See clause 5.4.5.3 |
| Source availability duration (ad) | The predicted duration for which corrections for the ranging source are expected to remain available, relative to the modified Z-count for the first measurement block 0 seconds 10 s 20 s 2 530 s $2\ 540\ \text{s} \leq \text{duration}$ Prediction of source availability duration not provided | 0 1 2 253 254 255 |
| For N measurement blocks (ranging from i=1 to i=N): Nmax = 12 | | |
| Ranging source ID (id) | This parameter defines the identity of the ranging source to which subsequent message block data are applicable. GPS satellite IDs (PRN) Reserved GLONASS satellite IDs Unused. SBAS satellite IDs (PRN) Unused | 1 to 36 37 38 to 61 62 to 119 120 to 138 139 to 255 |
| Issue of data (iod) | The issue of data associated with the ephemeris data used to determine pseudorange and range rate corrections. For GPS, IOD encode GPS IODE parameter For GLONASS, IOD encode GLONASS t_b parameter For SBAS, IOD | 255 |
| Pseudorange correction (prc) | The correction to the ranging source pseudorange transmitted to the airborne subsystem -327,67 m -327,66 m -327,65 m 0 m 0.01 m +327,66 m +327,67 m Unused | 0 1 2 32 767 32 768 65 533 65 534 65 535 |

| Data field | Definition | Encoded Value |
|------------------------------------|---|---|
| Range rate correction (rrc) | The rate of change of the pseudorange correction -32,767 m -32,766 m -32,765 m 0 m 0,001 m +32,766 m +32,767 m Unused | 0 1 2 32 767 32 768 65 533 65 534 65 535 |
| σ_{pr_gnd} (sd) | The standard deviation of a normal distribution associated with the signal in space contribution of the pseudorange error. 0 m 0,02 m 0,04 m 5,06 m 5,08 m Ranging source correction invalid | 0 1 2 253 254 255 |
| B1 (b1), B2 (b2), B3 (b3), B4 (b4) | The integrity parameters associated with the pseudorange corrections provided in the same measurement block (see ICAO Annex 10 [7] and GBAS MOPS [8], clause A.2.1) Bit 8: Positive Negative Bits 1 to 7: 0 m 0,05 m 6,34 m 6,35 m Reference receiver was not used to compute the pseudorange correction: | 0 1 0 1 126 127 1000 0000 |
| GNS-B Message CRC (gc) | See clause 5.4.5.1 | |

5.4.3 Message Type 2

| Requirement reference | |
|-----------------------|---|
| 5.4.3.1 | A station shall be capable of decoding a GNS-B message type 2 as defined in table 5.73. |
| 5.4.3.2 | Fields in the message type 2 shall be decoded as defined in table 5.74. |

Table 5.73: GNS-B message type 2 bit encoding

| Description | Octet | Bit number | | | | | | | |
|---|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| | 5 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| message block identifier (mbi) | 6 | mbi ₈ | mbi ₇ | mbi ₆ | mbi ₅ | mbi ₄ | mbi ₃ | mbi ₂ | mbi ₁ |
| GNS-B ID (gid) | 7 | gid ₈ | gid ₇ | gid ₆ | gid ₅ | gid ₄ | gid ₃ | gid ₂ | gid ₁ |
| | 8 | gid ₁₆ | gid ₁₅ | gid ₁₄ | gid ₁₃ | gid ₁₂ | gid ₁₁ | gid ₁₀ | gid ₉ |
| | 9 | gid ₂₄ | gid ₂₃ | gid ₂₂ | gid ₂₁ | gid ₂₀ | gid ₁₉ | gid ₁₈ | gid ₁₇ |
| message identifier (gmi) | 10 | 0 | 0 | 0 | 0 | 0 | gmi ₃ | gmi ₂ | gmi ₁ |
| message length (len) | 11 | len ₈ | len ₇ | len ₆ | len ₅ | len ₄ | len ₃ | len ₂ | len ₁ |
| GNS-B reference receivers (r) GNS-B accuracy designator letter (ac) GNS-B continuity/integrity designator (cid) | 12 | cid ₃ | cid ₂ | cid ₁ | res | ac ₂ | ac ₁ | r ₂ | r ₁ |
| local magnetic variation (v) | 13 | v ₈ | v ₇ | v ₆ | v ₅ | v ₄ | v ₃ | v ₂ | v ₁ |
| res | 14 | res | res | res | res | res | v ₁₁ | v ₁₀ | v ₉ |
| σ vert iono grad (vi) | 15 | vi ₈ | vi ₇ | vi ₆ | vi ₅ | vi ₄ | vi ₃ | vi ₂ | vi ₁ |
| refractivity index (i) | 16 | i ₈ | i ₇ | i ₆ | i ₅ | i ₄ | i ₃ | i ₂ | i ₁ |
| scale height (h) | 17 | h ₈ | h ₇ | h ₆ | h ₅ | h ₄ | h ₃ | h ₂ | h ₁ |
| refractivity uncertainty (u) | 18 | u ₈ | u ₇ | u ₆ | u ₅ | u ₄ | u ₃ | u ₂ | u ₁ |
| latitude (lat) | 19 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| | 20 | lat ₁₆ | lat ₁₅ | lat ₁₄ | lat ₁₃ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| | 21 | lat ₂₄ | lat ₂₃ | lat ₂₂ | lat ₂₁ | lat ₂₀ | lat ₁₉ | lat ₁₈ | lat ₁₇ |
| | 22 | lat ₃₂ | lat ₃₁ | lat ₃₀ | lat ₂₉ | lat ₂₈ | lat ₂₇ | lat ₂₆ | lat ₂₅ |
| longitude (lon) | 23 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| | 24 | lon ₁₆ | lon ₁₅ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| | 25 | lon ₂₄ | lon ₂₃ | lon ₂₂ | lon ₂₁ | lon ₂₀ | lon ₁₉ | lon ₁₈ | lon ₁₇ |
| | 26 | lon ₃₂ | lon ₃₁ | lon ₃₀ | lon ₂₉ | lon ₂₈ | lon ₂₇ | lon ₂₆ | lon ₂₅ |
| ellipsoid height (h) | 27 | h ₈ | h ₇ | h ₆ | h ₅ | h ₄ | h ₃ | h ₂ | h ₁ |
| | 28 | h ₁₆ | h ₁₅ | h ₁₄ | h ₁₃ | h ₁₂ | h ₁₁ | h ₁₀ | h ₉ |
| | 29 | h ₂₄ | h ₂₃ | h ₂₂ | h ₂₁ | h ₂₀ | h ₁₉ | h ₁₈ | h ₁₇ |
| GNS-B Message CRC (gc) | 30 | gc ₈ | gc ₇ | gc ₆ | gc ₅ | gc ₄ | gc ₃ | gc ₂ | gc ₁ |
| | 31 | gc ₁₆ | gc ₁₅ | gc ₁₄ | gc ₁₃ | gc ₁₂ | gc ₁₁ | gc ₁₀ | gc ₉ |
| | 32 | gc ₂₄ | gc ₂₃ | gc ₂₂ | gc ₂₁ | gc ₂₀ | gc ₁₉ | gc ₁₈ | gc ₁₇ |
| | 33 | gc ₃₂ | gc ₃₁ | gc ₃₀ | gc ₂₉ | gc ₂₈ | gc ₂₇ | gc ₂₆ | gc ₂₅ |

Table 5.74: GNS-B message type 2 field encoding

| Data field | Definition | Encoded Value |
|---|---|--|
| Message Identifier (gmi) | See table 5.70 | 2 |
| GNS-B message block identifier (mbi) | As defined in GNS-B message type 1 encoding | |
| GNS-B ID (gid) | As defined in GNS-B message type 1 encoding | |
| Message length (len) | As defined in GNS-B message type 1 encoding | |
| GNS-B reference receivers (r) | This defines the number of GNSS reference receivers installed: GNS-B installed with 2 reference receivers GNS-B installed with 3 reference receivers GNS-B installed with 4 reference receivers Reserved | 0 1 2 3 |
| GNS-B accuracy designator letter (ac) | This parameter defines the letter designator indicating the minimum signal-in-space accuracy performance provided by GNS-B: GNS-B has accuracy designation A GNS-B has accuracy designation B GNS-B has accuracy designation C Reserved | 0 1 2 3 |
| GNS-B continuity/integrity designator (cid) | The GNS-B Continuity / Integrity Designator (GCID) is a numerical designator which defines the operational performance of the GNS-B: Reserved GCID 1 GCID 2 GCID 3 GCID 4 Reserved Reserved GNS-B unhealthy | 0 1 2 3 4 5 6 7 |
| Local magnetic variation (v) | This data parameter defines the published magnetic variation at the GNS-B reference point: Bit 1: Positive, denotes eastward variation (clockwise from due north) Negative, denotes west variation (counter-clockwise from true north) Bits 2 to 11: 0 degrees 0,25 degrees 0,50 degrees 180 degrees Unused | 0 1 0 1 2 720 721 to 1023 |
| σ vert iono grad (vi) | This data parameter represents the standard deviation of a normal distribution associated with the residual ionospheric uncertainty due to spatial decorrelation 0 $0,1 \times 10^{-6}$ m/m $0,2 \times 10^{-6}$ m/m $25,5 \times 10^{-6}$ m/m | 0 1 2 255 |

| Data field | Definition | Encoded Value |
|------------------------------|---|--|
| Refractivity index (i) | This defines the refractivity index parameter value stored in the GNS-B Ground Subsystem. Bit 8: Positive Negative Bits 1 to 7: 0 3 378 381 | 0 1 0 1 126 127 |
| Scale height (h) | This defines the scale height parameter value stored in the GNS-B Ground Subsystem. 0 100 200 25 400 25 500 | 0 1 2 254 255 |
| Refractivity uncertainty (u) | This defines the refractivity uncertainty parameter value stored in the GNS-B Ground Subsystem. 0 1 2 255 | 0 1 2 255 |
| Latitude (lat) | This defines the GNS-B Reference Point latitude parameter value as stored in GNS-B Ground Subsystem. Bit 32: Positive (North) Negative (South) Bits 1 to 31: 0 degrees 0 minutes 0 arcseconds 0 degrees 0 minutes 0,0005 arcseconds 0 degree 0 minutes 0,00010 arcseconds 89 degrees 59 minutes 59,9995 arcseconds 90 degrees 0 minutes 0 arcseconds Unused | 0 1 0 1 2 647 999 999 648 000 000 648 000 001 to 2 147 483 647 |
| Longitude (lon) | This defines the GNS-B Reference Point longitude parameter value as stored in the GNS-B Ground Subsystem Bit 32: Positive (East) Negative (West) Bits 1 to 31: 0 degrees 0 minutes 0 arcseconds 0 degrees 0 minutes 0,0005 arcseconds 0 degree 0 minutes 0,00010 arcseconds 89 degrees 59 minutes 59,9995 arcseconds 90 degrees 0 minutes 0 arcseconds 180 degrees 0 minutes 0 arcseconds Unused | 0 1 0 1 2 647 999 999 648 000 000 1 296 000 000 1 296 000 001 to 2 147 483 647 |

| Data field | Definition | Encoded Value |
|------------------------|--|--|
| Ellipsoid height (h) | This defines the GNS-B reference point height above the WGS-84 ellipsoid parameter as stored in the GNS-B ground subsystem. Bit 24: Positive Negative Bits 1 to 23: 0 m 0.01 m 83 886.07 m | 0 1 0 1 8 388 607 |
| GNS-B Message CRC (gc) | See clause 5.4.5.1. | |

5.4.4 Message Type 4

| Requirement reference | |
|-----------------------|--|
| 5.4.4.1 | A station shall be capable of decoding a GNS-B message type 4 as defined in table 5.75. |
| 5.4.4.2 | Fields in the message type 4 shall be decoded as defined in table 5.76. |
| 5.4.4.3 | Message type 4 shall contain up to eight Final Approach Segment Data (FAS) blocks. |
| 5.4.4.3a | Each FAS data block shall be decoded according to ICAO Annex 10 [7] and clause A.2.4 in GBAS MOPS [8] if not specified otherwise here. |
| 5.4.4.4 | Void. |
| 5.4.4.5 | The GNS-B reference point shall be defined as the WGS-84 co-ordinates of a single RR antenna location for each installation. |

Table 5.75: GNS-B message type 4 bit encoding

| Description | Octet | Bit number | | | | | | | |
|---|-------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| | 5 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| message block identifier (mbi) | 6 | mbi ₈ | mbi ₇ | mbi ₆ | mbi ₅ | mbi ₄ | mbi ₃ | mbi ₂ | mbi ₁ |
| GNS-B ID (gid) | 7 | gid ₈ | gid ₇ | gid ₆ | gid ₅ | gid ₄ | gid ₃ | gid ₂ | gid ₁ |
| | 8 | gid ₁₆ | gid ₁₅ | gid ₁₄ | gid ₁₃ | gid ₁₂ | gid ₁₁ | gid ₁₀ | gid ₉ |
| | 9 | gid ₂₄ | gid ₂₃ | gid ₂₂ | gid ₂₁ | gid ₂₀ | gid ₁₉ | gid ₁₈ | gid ₁₇ |
| message identifier (gmi) | 10 | 0 | 0 | 0 | 0 | 0 | gmi ₃ | gmi ₂ | gmi ₁ |
| message length (len) | 11 | len ₈ | len ₇ | len ₆ | len ₅ | len ₄ | len ₃ | len ₂ | len ₁ |
| data set length (l) | 12 | l ₈ | l ₇ | l ₆ | l ₅ | l ₄ | l ₃ | l ₂ | l ₁ |
| For N FAS data blocks (i=1 to i=N): operation type (ot) SBAS provider ID (sp) | 13+(i-1)x40 | sp ₄ | sp ₃ | sp ₂ | sp ₁ | ot ₄ | ot ₃ | ot ₂ | ot ₁ |
| airport ID (ai) | 14+(i-1)x40 | ai ₈ | ai ₇ | ai ₆ | ai ₅ | ai ₄ | ai ₃ | ai ₂ | ai ₁ |
| | 15+(i-1)x40 | ai ₁₆ | ai ₁₅ | ai ₁₄ | ai ₁₃ | ai ₁₂ | ai ₁₁ | ai ₁₀ | ai ₉ |
| | 16+(i-1)x40 | ai ₂₄ | ai ₂₃ | ai ₂₂ | ai ₂₁ | ai ₂₀ | ai ₁₉ | ai ₁₈ | ai ₁₇ |
| | 17+(i-1)x40 | ai ₃₂ | ai ₃₁ | ai ₃₀ | ai ₂₉ | ai ₂₈ | ai ₂₇ | ai ₂₆ | ai ₂₅ |
| runway number (rn) runway letter (rl) | 18+(i-1)x40 | rl ₂ | rl ₁ | rn ₆ | rn ₅ | rn ₄ | rn ₃ | rn ₂ | rn ₁ |
| approach performance designator (apd) route indicator (ri) | 19+(i-1)x40 | ri ₅ | ri ₄ | ri ₃ | ri ₂ | ri ₁ | apd ₃ | apd ₂ | apd ₁ |
| reference path data selector (rps) | 20+(i-1)x40 | rps ₈ | rps ₇ | rps ₆ | rps ₅ | rps ₄ | rps ₃ | rps ₂ | rps ₁ |

| Description | Octet | Bit number | | | | | | | |
|--|-------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| reference path ID (rpi) | 21+(i-1)x40 | rpi ₈ | rpi ₇ | rpi ₆ | rpi ₅ | rpi ₄ | rpi ₃ | rpi ₂ | rpi ₁ |
| | 22+(i-1)x40 | rpi ₁₆ | rpi ₁₅ | rpi ₁₄ | rpi ₁₃ | rpi ₁₂ | rpi ₁₁ | rpi ₁₀ | rpi ₉ |
| | 23+(i-1)x40 | rpi ₂₄ | rpi ₂₃ | rpi ₂₂ | rpi ₂₁ | rpi ₂₀ | rpi ₁₉ | rpi ₁₈ | rpi ₁₇ |
| | 24+(i-1)x40 | rpi ₃₂ | rpi ₃₁ | rpi ₃₀ | rpi ₂₉ | rpi ₂₈ | rpi ₂₇ | rpi ₂₆ | rpi ₂₅ |
| LTP/FTP latitude (ltla) | 25+(i-1)x40 | ltla ₈ | ltla ₇ | ltla ₆ | ltla ₅ | ltla ₄ | ltla ₃ | ltla ₂ | ltla ₁ |
| | 26+(i-1)x40 | ltla ₁₆ | ltla ₁₅ | ltla ₁₄ | ltla ₁₃ | ltla ₁₂ | ltla ₁₁ | ltla ₁₀ | ltla ₉ |
| | 27+(i-1)x40 | ltla ₂₄ | ltla ₂₃ | ltla ₂₂ | ltla ₂₁ | ltla ₂₀ | ltla ₁₉ | ltla ₁₈ | ltla ₁₇ |
| | 28+(i-1)x40 | ltla ₃₂ | ltla ₃₁ | ltla ₃₀ | ltla ₂₉ | ltla ₂₈ | ltla ₂₇ | ltla ₂₆ | ltla ₂₅ |
| LTP/FTP longitude (ltlo) | 29+(i-1)x40 | ltlo ₈ | ltlo ₇ | ltlo ₆ | ltlo ₅ | ltlo ₄ | ltlo ₃ | ltlo ₂ | ltlo ₁ |
| | 30+(i-1)x40 | ltlo ₁₆ | ltlo ₁₅ | ltlo ₁₄ | ltlo ₁₃ | ltlo ₁₂ | ltlo ₁₁ | ltlo ₁₀ | ltlo ₉ |
| | 31+(i-1)x40 | ltlo ₂₄ | ltlo ₂₃ | ltlo ₂₂ | ltlo ₂₁ | ltlo ₂₀ | ltlo ₁₉ | ltlo ₁₈ | ltlo ₁₇ |
| | 32+(i-1)x40 | ltlo ₃₂ | ltlo ₃₁ | ltlo ₃₀ | ltlo ₂₉ | ltlo ₂₈ | ltlo ₂₇ | ltlo ₂₆ | ltlo ₂₅ |
| LTP/FTP height (lth) | 33+(i-1)x40 | lth ₈ | lth ₇ | lth ₆ | lth ₅ | lth ₄ | lth ₃ | lth ₂ | lth ₁ |
| | 34+(i-1)x40 | lth ₁₆ | lth ₁₅ | lth ₁₄ | lth ₁₃ | lth ₁₂ | lth ₁₁ | lth ₁₀ | lth ₉ |
| DFPAP latitude (fpla) | 35+(i-1)x40 | fpla ₈ | fpla ₇ | fpla ₆ | fpla ₅ | fpla ₄ | fpla ₃ | fpla ₂ | fpla ₁ |
| | 36+(i-1)x40 | fpla ₁₆ | fpla ₁₅ | fpla ₁₄ | fpla ₁₃ | fpla ₁₂ | fpla ₁₁ | fpla ₁₀ | fpla ₉ |
| | 37+(i-1)x40 | fpla ₂₄ | fpla ₂₃ | fpla ₂₂ | fpla ₂₁ | fpla ₂₀ | fpla ₁₉ | fpla ₁₈ | fpla ₁₇ |
| DFPAP longitude (fplo) | 38+(i-1)x40 | fplo ₈ | fplo ₇ | fplo ₆ | fplo ₅ | fplo ₄ | fplo ₃ | fplo ₂ | fplo ₁ |
| | 39+(i-1)x40 | fplo ₁₆ | fplo ₁₅ | fplo ₁₄ | fplo ₁₃ | fplo ₁₂ | fplo ₁₁ | fplo ₁₀ | fplo ₉ |
| | 40+(i-1)x40 | fplo ₂₄ | fplo ₂₃ | fplo ₂₂ | fplo ₂₁ | fplo ₂₀ | fplo ₁₉ | fplo ₁₈ | fplo ₁₇ |
| approach threshold crossing height (aph) | 41+(i-1)x40 | aph ₈ | aph ₇ | aph ₆ | aph ₅ | aph ₄ | aph ₃ | aph ₂ | aph ₁ |
| approach TCH units selector (aptu) | 42+(i-1)x40 | aptu ₁ | aph ₁₅ | aph ₁₄ | aph ₁₃ | aph ₁₂ | aph ₁₁ | aph ₁₀ | aph ₉ |
| glide path angle (gpa) | 43+(i-1)x40 | gpa ₈ | gpa ₇ | gpa ₆ | gpa ₅ | gpa ₄ | gpa ₃ | gpa ₂ | gpa ₁ |
| | 44+(i-1)x40 | gpa ₁₆ | gpa ₁₅ | gpa ₁₄ | gpa ₁₃ | gpa ₁₂ | gpa ₁₁ | gpa ₁₀ | gpa ₉ |
| course width (cw) | 45+(i-1)x40 | cw ₈ | cw ₇ | cw ₆ | cw ₅ | cw ₄ | cw ₃ | cw ₂ | cw ₁ |
| DLength offset (dlo) | 46+(i-1)x40 | dlo ₈ | dlo ₇ | dlo ₆ | dlo ₅ | dlo ₄ | dlo ₃ | dlo ₂ | dlo ₁ |
| FAS CRC (fc) | 47+(i-1)x40 | fc ₈ | fc ₇ | fc ₆ | fc ₅ | fc ₄ | fc ₃ | fc ₂ | fc ₁ |
| | 48+(i-1)x40 | fc ₁₆ | fc ₁₅ | fc ₁₄ | fc ₁₃ | fc ₁₂ | fc ₁₁ | fc ₁₀ | fc ₉ |
| | 49+(i-1)x40 | fc ₂₄ | fc ₂₃ | fc ₂₂ | fc ₂₁ | fc ₂₀ | fc ₁₉ | fc ₁₈ | fc ₁₇ |
| | 50+(i-1)x40 | fc ₃₂ | fc ₃₁ | fc ₃₀ | fc ₂₉ | fc ₂₈ | fc ₂₇ | fc ₂₆ | fc ₂₅ |
| FAS vertical alert limit /approach status (va) | 51+(i-1)x40 | va ₈ | va ₇ | va ₆ | va ₅ | va ₄ | va ₃ | va ₂ | va ₁ |
| FAS lateral alert limit/approach status (la) | 52+(i-1)x40 | la ₈ | la ₇ | la ₆ | la ₅ | la ₄ | la ₃ | la ₂ | la ₁ |
| GNS-B message CRC (gc) | 53+(i-1)x40 | gc ₈ | gc ₇ | gc ₆ | gc ₅ | gc ₄ | gc ₃ | gc ₂ | gc ₁ |
| | 54+(i-1)x40 | gc ₁₆ | gc ₁₅ | gc ₁₄ | gc ₁₃ | gc ₁₂ | gc ₁₁ | gc ₁₀ | gc ₉ |
| | 55+(i-1)x40 | gc ₂₄ | gc ₂₃ | gc ₂₂ | gc ₂₁ | gc ₂₀ | gc ₁₉ | gc ₁₈ | gc ₁₇ |
| | 56+(i-1)x40 | gc ₃₂ | gc ₃₁ | gc ₃₀ | gc ₂₉ | gc ₂₈ | gc ₂₇ | gc ₂₆ | gc ₂₅ |

Table 5.76: GNS-B message type 4 field encoding

| Data field | Definition | Encoded Value |
|--------------------------------------|---|---------------|
| Message Identifier (gmi) | See table 5.70 | 4 |
| GNS-B message block identifier (mbi) | As defined in GNS-B message type 1 encoding | |
| GNS-B ID (gid) | As defined in GNS-B message type 1 encoding | |
| Message length (len) | As defined in GNS-B message type 1 encoding | |

| Data field | Definition | Encoded Value |
|---------------------------------------|---|--|
| Data set length (l) | This field denotes the Type 4 Message data set length, which indicates the number of bytes in the data set. Unused 2 3 212 Unused | 0 to 1 2 3 212 212 to 255 |
| FAS data block (f) | See encoding for data fields below. | |
| Operation type (ot) | This defines the operation type parameter value as stored in the GNS-B Ground Subsystem and shall be set to the code applicable to "straight in approach" Straight in approach procedure Reserved | 0 1 to 15 |
| SBAS provider ID (sp) | This defines the SBAS service provider ID parameter value as stored in the GNS-B Ground Subsystem. NOTE: This parameter is not used for approaches conducted using the GNS-B Ground Subsystem corrections. WAAS EGNOS MSAS Reserved FAS data block is to be used with GBAS only. FAS data block can be used with any SBAS service provider. | 0 1 2 3 to 13 14 15 |
| Airport ID (ai) | This defines the airport identification parameter value as stored in the GNS-B Ground Subsystem. Each character is coded using the lower 6 bits of its IA-5 representation. For each character b_i is transmitted first, and 2 zero bits are appended after b_6 , so that 8 bits are transmitted for each character. Only upper case letters, numeric digits and IA-5 "space" are used. The rightmost character is transmitted first. For a three-character GBAS ID, the rightmost (first transmitted) character shall be IA-5 "space". | |
| Runway number (rn) | This defines the runway number parameter value as stored in the GNS-B Ground Subsystem Heliport Runway number | 0 1 to 36 |
| Runway letter (rl) | This defines the runway letter parameter value as stored in the GNS-B Ground Subsystem No letter R (right) C (centre) L (left) | 0 1 2 3 |
| Approach performance designator (apd) | This defines the code applicable to Category I approach Reserved Category I Reserved for Category II Reserved for Category III Reserved | 0 1 2 3 4 to 7 |

| Data field | Definition | Encoded Value |
|------------------------------------|---|--|
| Route indicator (ri) | This defines the route indicator parameter value as stored in the GNS-B Ground Subsystem. The letter is coded using bits b_1 through b_5 of its IA-5 representation. Bit b_1 is transmitted first. Only upper case letters, excluding "I" and "O", or IA-5 "space" are used. | |
| Reference path data selector (rps) | This defines the reference path data selector parameter value as stored in the GNS-B Ground Subsystem 0 1 48 Unused | 0 1 48 49 to 255 |
| Reference path ID (rpi) | This defines the reference path identifier parameter value as stored in the GNS-B Ground Subsystem Each character is coded using bits b_1 through b_6 of its IA-5 representation. For each character, b_1 is transmitted first, and 2 zero bits are appended after b_6 so that 8 bits are transmitted for each character. Only upper case letters, numeric digits and IA-5 "space" are used. The rightmost character is transmitted first. For a three-character reference path identifier, the rightmost (first transmitted) character shall be IA-5 "space". | |
| LTP/FTP latitude (Itla) | This defines the LTP/FTP latitude parameter value as stored in the GNS-B Ground Subsystem and shall represent the latitude (in arcseconds) of the LTP/FTP point defined in WGS-84 co-ordinates Bit 32: Positive (North) Negative (South) Bits 1 to 31: 0 degrees 0 minutes 0 arcseconds 0 degrees 0 minutes 0,0005 arcseconds 0 degree 0 minutes 0,00010 arcseconds 89 degrees 59 minutes 59,9995 arcseconds 90 degrees 0 minutes 0 arcseconds Unused | 0 1 0 1 2 647 999 999 648 000 000 648 000 001 to 2 147 483 647 |
| LTP/FTP longitude (Itlo) | This defines the LTP/FTP longitude parameter value as stored in the GNS-B Ground Subsystem and shall represent the longitude (in arcseconds) of the LTP/FTP point defined in WGS-84 co-ordinates Bit 32: Positive (East) Negative (West) Bits 1 to 31: 0 degrees 0 minutes 0 arcseconds 0 degrees 0 minutes 0,0005 arcseconds 0 degree 0 minutes 0,00010 arcseconds 89 degrees 59 minutes 59,9995 arcseconds 90 degrees 0 minutes 0 arcseconds 180 degrees 0 minutes 0 arcseconds Unused | 0 1 0 1 2 647 999 999 648 000 000 1 296 000 000 1 296 000 001 to 2 147 483 647 |

| Data field | Definition | Encoded Value |
|---|---|--|
| LTP/FTP height (lth) | This defines the LTP/FTP height parameter value as stored in the GNS-B Ground Subsystem. -512,0 m -511,9 m 0,0 m 0,1 m 6 041,5 m | 0 1 5 120 5 121 65 535 |
| DFPAP latitude (fpla) | The delta Flight Path Alignment Point (FPAP) Latitude field defines the delta FPAP latitude parameter value as stored in the GNS-B Ground Subsystem. Bit 24 Positive Negative Bits 1 to 23 0 degrees 0 minutes 0 arcseconds 0 degrees 0 minutes 0,0005 arcseconds 1 degree 0 minutes 0 arcseconds Unused | 0 1 0 1 7 200 000 7 200 001 to 8 388 607 |
| DFPAP longitude (fplo) | The delta Flight Path Alignment Point (FPAP) Longitude field defines the delta FPAP longitude parameter value as stored in the GNS-B Ground Subsystem. Bit 24 Positive Negative Bits 1 to 23 0 degrees 0 minutes 0 arcseconds 0 degrees 0 minutes 0,0005 arcseconds 1 degree 0 minutes 0 arcseconds Unused | 0 1 0 1 7 200 000 7 200 001 to 8 388 607 |
| Approach threshold crossing height (apth) | The approach Threshold Crossing Height (TCH) field defines the TCH parameter value as stored in the GNS-B Ground Subsystem. In Metres: 0 m 0,05 m 1 638,35 m Alternatively In Feet: 0 ft 0,1 ft 3 276,7 ft | 0 1 32 767 0 1 32 767 |
| Approach TCH units selector (aptu) | This defines the TCH Unit Selector parameter value as stored in the GNS-B Ground Subsystem. Feet Metres | 0 1 |

| Data field | Definition | Encoded Value |
|---|--|--|
| Glide path angle (gpa) | The Glide Path Angle (GPA) field defines the GPA parameter value as stored in the GNS-B Ground Subsystem and shall represent the glidepath angle of the FAS path with respect to the horizontal plane tangent to the WGS-84 ellipsoid at the LTP/FTP. 0 degrees 0,01 degrees 0,02 degrees 89,99 degrees | 0 1 2 8 999 |
| | 90 degrees Unused | 9 000 9 001 to 65 535 |
| Course width (cw) | This defines the course width parameter value as stored in the GNS-B Ground Subsystem and shall represent the lateral displacement at which full-scale deflection of a course deviation indicator from the path defined by the FAS at the LTP/FTP is attained. 80 m 80,25 m 80,50 m 143,75 m | 0 1 2 255 |
| | DLength offset (dlo) | This defines the delta length offset parameter value as stored in the GNS-B Ground Subsystem and shall represent the distance from the stop end of the runway to the FAS Path Alignment Point. 0 m 8 m 16 m 24 m 2 032 m Not provided |
| Final Approach Segment CRC (fc) | This defines the FAS CRC value as stored in the GNS-B Ground Subsystem. The 32 bit cyclic redundancy check is appended to the end of each FAS data block in order to ensure FAS data integrity. The length of the CRC shall be k=32 bits. For encoding see clause 5.5.5.2 | |
| FAS vertical alert limit/approach status (va) | Vertical Alert Limit 0 m 0,1 m 0,2 m 25,4 m Do not use vertical deviations. | 0 1 2 254 255 |
| | FAS lateral alert limit/approach status (la) | Lateral Alert Limit 0 m 0,2 m 0,4 m 50,8 m Do not use approach |
| GNS-B Message CRC (gc) | See clause 5.4.5.1 | |

5.4.5 CRC Calculation

5.4.5.1 GNS-B CRC

| Requirement reference | |
|-----------------------|--|
| 5.4.5.1.1 | The GNS-B CRC generator polynomial shall be: $G(x) = x^{32} + x^{31} + x^{24} + x^{22} + x^{16} + x^{14} + x^8 + x^7 + x^5 + x^3 + x + 1$ |
| 5.4.5.1.2 | The CRC information field, $M(x)$, shall be: $M(x) = \sum_{i=1}^n m_i x^{n-i} = m_1 x^{n-1} + m_2 x^{n-2} + \dots + m_n x^0$ |
| 5.4.5.1.3 | $M(x)$ shall be formed from the 48 bit GNS-B message block header and all bits of the variable-length message, excluding the CRC. |
| 5.4.5.1.4 | Bits shall be arranged in the order transmitted, such that m_1 corresponds to the first transmitted bit of the message block header, and m_n corresponds to the last transmitted bit of the (n-48) message bits. |
| 5.4.5.1.5 | The following variables shall be used to compute the CRC-value. Start Value shall be 0. Reflect on input shall be done. Reflect on output shall be done. XOR on output shall not be done (XOR value shall be 0). |

5.4.5.2 FAS CRC

| Requirement reference | |
|-----------------------|--|
| 5.4.5.2.1 | The FAS CRC generator polynomial shall be $G(x) = x^{32} + x^{31} + x^{24} + x^{22} + x^{16} + x^{14} + x^8 + x^7 + x^5 + x^3 + x + 1$ |
| 5.4.5.2.2 | The CRC information field, $M(x)$, shall be: $M(x) = \sum_{i=1}^{272} m_i x^{272-i} = m_1 x^{271} + m_2 x^{270} + \dots + m_{272} x^0$ |
| 5.4.5.2.3 | $M(x)$ shall be formed from all bits of the associated FAS data block, excluding the CRC. |
| 5.4.5.2.4 | Bits shall be arranged in the order transmitted, such that m_1 corresponds to the LSB of the operation type field, and m_{272} corresponds to the MSB of the Δ Length Offset. |
| 5.4.5.2.5 | The following variables shall be used to compute the CRC-value. Start Value shall be 0. Reflect on input shall be done. Reflect on output shall be done. XOR on output shall not be done (XOR value shall be 0). |

5.4.5.3 Ephemeris CRC

| Requirement reference | |
|-----------------------|---|
| 5.4.5.3.1 | The "Ephemeris CRC" field shall contain the Ephemeris CRC computed for the ranging source associated with the first ranging source measurement block in the Type 1 Message. |
| 5.4.5.3.2 | For a SBAS geostationary satellite, the Ephemeris CRC shall be coded as all zeros. |
| 5.4.5.3.3 | The ephemeris CRC generator polynomial shall be: $G(x) = x^{16} + x^{12} + x^5 + 1$ |
| 5.4.5.3.4 | The CRC information field, $M(x)$, for a given satellite shall be: $M(x) = \sum_{i=1}^n m_i x^{n-i} = m_1 x^{n-1} + m_2 x^{n-2} + \dots + m_n x^0$ |
| 5.4.5.3.5 | For a GPS satellite, $M(x)$ shall be of length $n=576$ bits. |
| 5.4.5.3.6 | $M(x)$ for a GPS satellite shall be calculated using the first 24 bits from each of Words 3 through 10 of subframes 1, 2 and 3 of the data transmission from that satellite, ANDed with the GPS satellite ephemeris mask shown in table 5.77. |
| 5.4.5.3.7 | After the AND operation, $M(x)$ shall be arranged in the order that bytes are transmitted by the GPS satellite, but with each byte ordered least-significant bit first, such that m_1 corresponds to bit 68 of subframe 1 (LSB of subframe 1, word 3), and m_{576} corresponds to bit 287 of subframe 3 (MSB of subframe 3, word 10). |
| 5.4.5.3.8 | The CRC shall be ordered such that r_1 is the first bit transmitted and r_{16} is the last bit transmitted. |
| 5.4.5.3.9 | The following variables shall be used to compute the CRC-value. Start Value shall be 0. Reflect on input shall be done. Reflect on output shall be done. XOR on output shall not be done (XOR value shall be 0). |

Table 5.77: GPS Satellite ephemeris mask

| Subframe 1: | Byte 1 | Byte 2 | Byte 3 | | Byte 1 | Byte 2 | Byte 3 |
|-------------|-----------|-----------|-----------|---------|-----------|-----------|-----------|
| Word 3 | 0000 0000 | 0000 0000 | 0000 0011 | Word 4 | 0000 0000 | 0000 0000 | 0000 0000 |
| Word 5 | 0000 0000 | 0000 0000 | 0000 0000 | Word 6 | 0000 0000 | 0000 0000 | 0000 0000 |
| Word 7 | 0000 0000 | 0000 0000 | 1111 1111 | Word 7 | 1111 1111 | 1111 1111 | 1111 1111 |
| Word 9 | 1111 1111 | 1111 1111 | 1111 1111 | Word 10 | 1111 1111 | 1111 1111 | 1111 1100 |
| Subframe 2: | Byte 1 | Byte 2 | Byte 3 | | Byte 1 | Byte 2 | Byte 3 |
| Word 3 | 1111 1111 | 1111 1111 | 1111 1111 | Word 4 | 1111 1111 | 1111 1111 | 1111 1111 |
| Word 5 | 1111 1111 | 1111 1111 | 1111 1111 | Word 6 | 1111 1111 | 1111 1111 | 1111 1111 |
| Word 7 | 1111 1111 | 1111 1111 | 1111 1111 | Word 7 | 1111 1111 | 1111 1111 | 1111 1111 |
| Word 9 | 1111 1111 | 1111 1111 | 1111 1111 | Word 10 | 1111 1111 | 1111 1111 | 0000 0000 |
| Subframe 3: | Byte 1 | Byte 2 | Byte 3 | | Byte 1 | Byte 2 | Byte 3 |
| Word 3 | 1111 1111 | 1111 1111 | 1111 1111 | Word 4 | 1111 1111 | 1111 1111 | 1111 1111 |
| Word 5 | 1111 1111 | 1111 1111 | 1111 1111 | Word 6 | 1111 1111 | 1111 1111 | 1111 1111 |
| Word 7 | 1111 1111 | 1111 1111 | 1111 1111 | Word 7 | 1111 1111 | 1111 1111 | 1111 1111 |
| Word 9 | 1111 1111 | 1111 1111 | 1111 1111 | Word 10 | 1111 1111 | 1111 1111 | 1111 1100 |

6 General design requirements

The equipment shall meet the requirements of EN 302 842-2 [5] clause 6.

7 Protocol test procedures

7.1 General

The equipment shall meet the requirements of EN 302 842-2 [5] clause 7.1.

7.2 Required test rig

An overview of the PCOs identified as required for the conduct of protocol test cases is given in EN 302 842-2 [5] clause 7.2.

7.3 Protocol test-suite description methodology

The equipment shall meet the requirements of EN 302 842-2 [5] clause 7.3.

7.4 Detailed protocol test procedures

The test procedures set forth below constitute a satisfactory method of determining the required VDL Mode 4 mobile station performance. Although specific test procedures are cited, it is recognized that other methods may be preferred. Such alternate methods may be used if the manufacturer can show that they provide at least equivalent information. Therefore, the procedures cited herein should be used as one criterion in evaluating the acceptability of the alternate procedures.

7.4.1 Test-suite overview

The test-suite overview shown in table 7.1 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 |
|---------------------|---|
| ADSB_Request_A | To demonstrate that a station which desires another station to transmit a single autonomous synchronization burst will transmit an ADS-B request burst with the sleep and auto bits set to zero. |
| ADSB_Request_B | To demonstrate that a station which desires another station to transmit a single autonomous synchronization burst with a specified variable part will transmit an ADS-B request burst with sleep=0 and auto=1 and include the auto parameters. |
| ADSB_Request_C | To demonstrate that a station which desires another station to transmit synchronization bursts autonomously with the use of sleep mode, will transmit an ADS-B request burst with sleep=1 and auto=0 and include the sleep parameters. |
| ADSB_Request_D | To demonstrate that a station which desires another station to transmit synchronization bursts with a specified variable part and with the use of sleep mode, will transmit an ADS-B request burst with sleep=1 and auto=1 and include the auto and sleep parameters. |
| ADSB_Reporting | To demonstrate that a station will send at least 12 sync bursts a minute averaged over the two GSCs. |
| ADSB_Sleep_A | To demonstrate that the sleep mode parameters are obeyed according to the position and velocity of the station under test, and that the sleep mode is overridden by a directed request. |
| ADSB_Sleep_B | To demonstrate that the position and velocity sleep mode parameters are ignored when either pos or vel, respectively, are zero. |
| ADSB_Request_Time | To demonstrate that a station requesting the time synchronization information field (information field ID 2) will only transmit a request to a station that has announced that it is operating with a primary time source (tfom = 0 or 1). |
| ADSB_Priority | To demonstrate that information fields 0, 1, 2, 3, 4 and A1 hex have priority over other information fields. |
| ADSB_First_TCP_A | To demonstrate that when the status of its first TCP changes, a mobile station will autonomously transmit a series of synchronization bursts containing the single-slot TCP variable part. |
| ADSB_Status_SVQ_A | To demonstrate that when the status of its SVQ changes, a mobile station will autonomously transmit a series of synchronization bursts containing the single-slot SVQ variable part. |
| ADSB_TCP/SVQ_Flag_A | To demonstrate that a station which is transmitting a series of directed synchronization bursts that include the two slot TCP/SVQ variable part will indicate when there is a change to any of its TCPs by setting the TCP/SVQ change flag to zero in all directed synchronization bursts transmitted by the station. |

| Test Case Name | Description |
|-----------------------------|---|
| ADSB_TCP/SVQ_Flag_B | To show that a station which is transmitting a series of directed synchronization bursts which do not include the two slot TCP/SVQ variable part will set the TCP/SVQ change flag to 1. |
| ADSB_Basic_A | To demonstrate that a station will broadcast a basic sync burst message with the correct format. |
| ADSB_Basic_Rec_A | To demonstrate that a station will correctly process a received basic sync burst. |
| ADSB_High_Dynamic_A | To demonstrate that a station will broadcast a high dynamic sync burst with the correct format. |
| ADSB_High_Dynamic_Rec_A | To demonstrate that a station will correctly process a received high dynamic sync burst. |
| ADSB_Full_Position_A | To demonstrate that a station will broadcast a full position sync burst with the correct format. |
| ADSB_Full_Position_Rec_A | To demonstrate that a station will correctly process a received full position sync burst. |
| ADSB_Basic_Ground_Rec_A | To demonstrate that a station will correctly process a received basic ground sync burst. |
| ADSB_UTC_Time_Rec_A | To demonstrate that a station will correctly process a received UTC time sync burst. |
| ADSB_Two_Slot_TCP/SVQ_A | To demonstrate that a station will broadcast a two slot TCP/SVQ sync burst with the correct format. |
| ADSB_Two_Slot_TCP/SVQ_Rec_A | To demonstrate that a station will correctly process a received two slot TCP/SVQ sync burst. |
| ADSB_Single_Slot_TCP_A | To demonstrate that a station will broadcast a single slot TCP sync burst with the correct format. |
| ADSB_Single_Slot_TCP_Rec_A | To demonstrate that a station will correctly process a received single slot TCP sync burst. |
| ADSB_Single_Slot_SVQ_A | To demonstrate that a station will broadcast a single slot SVQ sync burst with the correct format. |
| ADSB_Single_Slot_SVQ_Rec_A | To demonstrate that a station will correctly process a received single slot SVQ sync burst. |
| ADSB_Aircraft_Data_A | To demonstrate that a station will broadcast an aircraft data sync burst with the correct format. |
| ADSB_Aircraft_Data_Rec_A | To demonstrate that a station will correctly process a received aircraft data sync burst. |
| ADSB_High_Resolution_A | To demonstrate that a station will broadcast a high resolution sync burst with the correct format. |
| ADSB_High_Resolution_Rec_A | To demonstrate that a station will correctly process a received high resolution sync burst. |
| Channel_Dest_Count | To demonstrate that if the destination count (dc) in the header block is zero, then the channel management CTRL will apply to all stations and the receiving station will operate in directed rate mode. |
| Channel_Dir_Rate_A | To demonstrate that a mobile station receiving a channel management parameter with the directed offset (do) equal to zero in the header block and/or with the individual offset (ido) set equal to zero in a destination block with the station's address, with styp = 0, will operate in directed rate mode. |
| Channel_Dir_Rate_B | To demonstrate that a mobile station receiving a channel management parameter with the directed offset (do) equal to zero in the header block and/or with the individual offset (ido) set equal to zero in a destination block with the station's address, with styp = 1, will operate in directed rate mode, with transmissions sharing streams in successive minutes. |
| Channel_Address | To demonstrate that if none of the addresses matches the receiver, then further processing of the channel management CTRL will terminate. |
| Channel_Directed | To demonstrate that if the individual offset (ido) is not set equal to zero in a destination block with the station's address, then the addressed stations will operate in directed slot mode. |
| Channel_Multicast_A | To demonstrate that stations operating in directed-slot mode will not process regional multicast commands. |
| Channel_Multicast_B | To demonstrate that a mobile station will act on a regional multicast channel management CTRL if it is in the correct region (according to Condition 1), and that processing of the channel management CTRL will terminate if it is not. |
| Channel_Multicast_C | To demonstrate that a mobile station will act on a regional multicast channel management CTRL if it is in the correct region (according to Condition 2a), and that processing of the channel management CTRL will terminate if it is not. |

| Test Case Name | Description |
|------------------------|---|
| Channel_Multicast_D | To demonstrate that a mobile station will act on a regional multicast channel management CTRL if it is in the correct region (according to Condition 2b), and that processing of the channel management CTRL will terminate if it is not. |
| Channel_Invalid | To demonstrate that if styp=1 and NES is not a multiple of (2 + sr), then a channel management CTRL will be judged invalid, and the entire CTRL discarded without further processing. |
| Channel_Frequency | To demonstrate that if all of the elements that share a stream are not transmitted on the same frequency in a channel management CTRL, then the message will be judged invalid and the entire CTRL will be discarded. |
| Channel_Length | To demonstrate that if all of the elements that share a stream are not of the same length in a channel management CTRL, then a basic sync burst will be transmitted for all elements in that stream. |
| Channel_Plea | To demonstrate that if a station receives a plea request while transmitting according to a channel management CTRL, it shall use NTM as the default reporting rate for the frequency to construct the plea response. |
| Channel_Cancel_CSID_A | To demonstrate that a station previously in receipt of a channel management CTRL that is commanded to cancel all reservations for sync burst transmission by the same channel management CTRL but with csid = 15 will revert to default sync burst operations. |
| Channel_Cancel_CSID_B | To demonstrate that a station previously in receipt of a channel management CTRL that is commanded to cancel all reservations for sync burst transmission by a different channel management CTRL with csid = 15 will revert to default sync burst operations. |
| Channel_Cancel_Auto | To demonstrate that if a receiving station is transmitting autonomously the data for which a channel management command was received, then it will cancel its existing reservations and operate in accordance with the parameters of the CTRL command. |
| Channel_Cancel_CSID_C | To demonstrate that a station that receives a channel management CTRL parameter with csid = 0 will terminate any previous default sync burst operations and initiate operations in accordance with the current CTRL parameter. |
| Channel_Cancel_CSID_D | To demonstrate that a station that receives a channel management CTRL parameter with csid matching a previously-received channel management CTRL parameter will terminate any previous operations commanded by the previous CTRL parameter and initiate operations in accordance with the current CTRL parameter. |
| Channel_Unrecognize | To demonstrate that if a station receives a channel management parameter containing a standard transmission definition (id = 0 through 15) that it does not recognize, it will transmit a basic sync burst. |
| Channel_User_Trans | To demonstrate that a station will correctly interpret a channel management parameter with user-defined transmission blocks. |
| Channel_Non_Sync_Trans | To demonstrate that a station will correctly interpret a channel management parameter with non-sync burst transmission blocks. |
| Channel_Speed | To demonstrate that if a mobile is travelling greater than 3 069 knots, then it will transmit a high dynamic variable field whenever it otherwise would have transmitted a basic, full position, or high resolution variable field. |
| Channel_NotSupported | To demonstrate that whenever a mobile cannot transmit the information requested by a channel management CTRL, then the mobile will transmit a sync burst with a basic variable field. |
| Channel_Script_Repeat | To demonstrate that a station will correctly interpret a channel management parameter with repeat count in the script block. |
| Channel_Param_Q1 | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the Q1 parameter. |
| Channel_Param_Q4 | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the Q4 parameter. |
| Channel_Param_Q2 | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the Q2 parameters. |
| Channel_Param_V12 | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the V12 parameter. |
| Channel_Param_VS4 | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the VS4 parameter. |
| Channel_Param_VS2 | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the VS2 parameter. |
| Channel_Discard | To demonstrate that a receiving station will discard a channel management CTRL command containing a script block with two consecutive octets having bits 6 to 8 equal to "111". |

| Test Case Name | Description |
|-------------------------|---|
| CPR_Encode_Offset | To demonstrate that offset latitude and longitude positions are correctly encoded in the sync burst information fields using the CPR algorithm. |
| CPR_Decode_Offset | To demonstrate that offset latitude and longitude positions are correctly decoded from the sync burst information fields using the CPR algorithm. |
| TIS_TIV_Unique_A | To demonstrate that the same TIV ID in target messages referring to the same aircraft, but received from different ground stations, are identified as separate TIVs. |
| TIS_TIV_Unique_B | To demonstrate that the same TIV ID in target messages referring to different aircraft, and received from different ground stations, are identified as separate TIVs. |
| TIS_TIV_Unique_C | To demonstrate that the same TIV ID in target messages referring to different aircraft or vehicles on the ground, and received from different ground stations, are identified as separate TIVs. |
| TIS_Manage_Rec_A | To demonstrate that a station will correctly process a received management message. |
| TIS_Air_Air_Rec_A | To demonstrate that a station will correctly process a received aircraft airborne target message. |
| TIS_Air_Ground_Rec_A | To demonstrate that a station will correctly process a received aircraft target ground message. |
| TIS_Ground_Veh_Rec_A | To demonstrate that a station will correctly process a received ground vehicle target message. |
| FIS_Request_A | To demonstrate that a station will transmit a FIS-B report request message with the null reservation field with the correct format. |
| FIS_Request_B | To demonstrate that a station will transmit a FIS-B report request message with the unicast reservation field with the correct format. |
| FIS_METAR_Rec_A | To demonstrate that a station will correctly process a received METAR message. |
| FIS_SPECI_Rec_A | To demonstrate that a station will correctly process a received SPECI message. |
| FIS_ATIS_Rec_A | To demonstrate that a station will correctly process a received ATIS message. |
| FIS_RVR_Rec_A | To demonstrate that a station will correctly process a received RVR message. |
| FIS_SIGMET_Rec_A | To demonstrate that a station will correctly process a received SIGMET message. |
| GNS_Type1_Message_Rec_A | To demonstrate that a station will correctly process a received GNS-B Type 1 message. |
| GNS_Type2_Message_Rec_A | To demonstrate that a station will correctly process a received GNS-B Type 2 message. |
| GNS_Type4_Message_Rec_A | To demonstrate that a station will correctly process a received GNS-B Type 4 message. |

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

Abbreviations and other terms for protocol test cases are given in EN 302 842-2 [5] clause 7.4.3.1 unless stated in table 7.2.

7.4.3.1.1 Subfield mnemonics

Table 7.2: Subfield mnemonics

| Mnemonic | Meaning |
|----------|--------------------------|
| a | Additional slots |
| aavl | Airport availability |
| ac | Aircraft category |
| a/d | Autonomous/directed flag |
| adsff | ADS-B fault flag |
| aflg | Altitude resolution flag |

| Mnemonic | Meaning |
|-----------------|---------------------------------|
| ai | Additional service Information |
| altr | Altitude rate |
| anum | Application number |
| aptf | Approach type flag |
| apty | Approach type |
| atis | No of ADS-B targets |
| auto | Autonomous information |
| aux | Auxiliary selection |
| balt | Base altitude |
| bar | Barometric altitude |
| b/g | Baro/geo altitude |
| bgo | Baro/geo offset |
| blg | Block length |
| bo | Block offset |
| br | Block repeat rate |
| br | Baro rate |
| brk | Braking action |
| bs | Block start |
| bt | Block timeout |
| c | CRC |
| call | Callsign |
| cat | Aircraft category |
| cav | CAVOK flag |
| cc | Cloud coverage |
| ch | Cloud height |
| cno | Cloud number flag |
| cprf | CPR format even/odd |
| crp | Phenomenon / condition reported |
| csid | Command set ID |
| csl | Call sign left |
| csr | Call sign right |
| cty | Cloud type |
| d | Destination address |
| da | Data age |
| date | Date |
| day | UTC day |
| dc | Directory count |
| dew | Dew point temperature |
| di | Date issued |
| DLPDU | Data Link Protocol Data Unit |
| do | Directed offset |
| dod | Depth of deposit |
| dos | Directory of services flag |
| dt | Directed timeout |
| dvt | Date valid to |
| erid | Extended reservation ID |
| f | Frequency |
| fc | Frequency count |
| fidflg | Flight ID flag |
| fidty | Flight ID type |
| flag | Flag delimiting burst |
| fmi | FIS-B message ID |
| freq | Frequency |
| frq | Frequency index |
| ftxt | Free text flag |
| gr | Geo rate |
| gs | Ground speed |
| gsc | Global Signalling Channel (GSC) |
| gsp | Ground speed |
| gt | Ground track |
| gtk | Ground track |
| h | UTC hour |
| hd | Holding delay |

| Mnemonic | Meaning |
|-----------------|---|
| id | Information field identity |
| ido | Individual offset |
| in | Information field |
| io | Incremental offset |
| lat | Latitude |
| lat4 | 4-bit latitude offset |
| lat6 | 6-bit latitude offset |
| lat8 | 8-bit latitude offset |
| lbar | Lower barometric altitude |
| lg | Length |
| loci | Location indicator |
| locl | Local ID |
| lon | Longitude |
| lon4 | 4-bit longitude offset |
| lon6 | 6-bit longitude offset |
| lon8 | 8-bit longitude offset |
| long | Reference point longitude |
| mas | Maximum visibility |
| masd | Maximum visibility direction |
| med | Message edition |
| mi | Message ID |
| min | UTC minute |
| mis | Minimum visibility |
| misd | Minimum visibility direction |
| mon | UTC month |
| mrq | Message requested |
| mxw | max wind flag |
| n | CTRL length |
| nd | Negative dither |
| no | TCP number |
| nr | Nominal update rate |
| nsd | Non-standard definition |
| ntis | No of TIS-B targets |
| nic | Position navigation integrity category |
| nvert | Number of vertices |
| off | Offset to first reserved slot |
| ok | Confirm/failure flag |
| or | Override flag |
| ori | Originator indicator |
| oth | Other operational activity |
| pc | Parameter count |
| pid | Patch ID |
| po | Periodic offset |
| pos | Position |
| pr | Priority |
| pr_flag | Plea response flag |
| prm | VSS user specific parameter |
| pt | Periodic timeout |
| pw | Present weather |
| pwf | Present weather flag |
| Q1 | Priority |
| Q2a | Slot selection range constraint for level n |
| Q2b | Slot selection range constraint for level n |
| Q2c | Slot selection range constraint for level n |
| Q2d | Slot selection range constraint for level n |
| Q4 | Number of available slots |
| qfe | QFE |
| qfef | QFE flag |
| qnh | QNH |
| ravl | Runway availability |
| r-b/a | Requested baro/geo altitude |
| rlat | Latitude |
| r-mi | Requested message ID |

| Mnemonic | Meaning |
|-----------------|-------------------------------------|
| ro | Response offset |
| roff | Re-broadcast offset |
| rso | RVR source |
| rcon | Runway contamination |
| rcvr | Receiver control |
| rd | Reservation data |
| rdp | Runway deposits |
| res | Reserved bit |
| rflg | Radar fusion flag |
| rri | Reduced runway length |
| rrw | Reduced runway width |
| rid | Reservation ID |
| r-id | Requested information ID subfield |
| rlg | Longitude |
| ro | Response offset |
| roff | Re-broadcast offset |
| ruf | Runway flag |
| run | Runway ID |
| rus | Runway in use |
| rve | RVR endpoint |
| rvm | RVR midpoint |
| rvt | RVR touchdown |
| rw | Recent weather |
| rwf | Recent weather flag |
| s | Source address |
| sc | Script count |
| sdf | Source/destination flag |
| sec | UTC second |
| sfl | SIGMET flag |
| sleep | Autonomous monitoring |
| si | Service information |
| sil | Surveillance integrity level |
| sit | Service information type |
| slt | Slot |
| snr | Secondary reporting rate |
| sqn | FIS-B sequence number |
| sr | Script rate |
| ss | State of the sea |
| ssf | Sea flag |
| sst | Sea surface temperature |
| st | Status |
| styp | Script duration type |
| sz | Size |
| tacc | TIS-B target resolution |
| tag | Time stamp |
| tem | Temperature |
| tqc | TCP/SVQ change flag |
| tflg | Target identifier flag |
| tfom | Time figure of merit |
| ti | Time |
| tid | Target Identifier |
| tind | Turn indication |
| tivid | TIV ID |
| tmi | TIS-B message ID |
| tran | Transition time |
| trd | TREND report flag |
| trmt | Transmit control |
| ttg | Time to go |
| TV11max | Reservation hold time maximum value |
| TV11min | Reservation hold time minimum value |
| tv | Time valid to |
| txd | Transmission definition index |
| txt | Free text message |

| Mnemonic | Meaning |
|----------|------------------------------|
| typ | TCP type |
| ubar | Upper barometric altitude |
| ucid | UCTRL identity |
| up | Update period |
| vdif | Variable wind direction from |
| vdit | Variable wind direction to |
| vel | Maximum sleep velocity |
| ver | Version number |
| vers | FIS-B version ID |
| vlat | Vertex latitude |
| vlg | Vertex longitude |
| vt | Timeout |
| V11 | Nominal periodic rate |
| wdi | Wind direction |
| wgi | Wind gust indicator |
| ws | Wind shear |
| wsf | Wind shear flag |
| wsp | Wind speed |
| yr | UTC year |

7.4.3.1.2 Station addresses and positions

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

- add_A = address of the station under test (station A);
- add_B = address of simulated station B (simulated by the test equipment);
- add_C = address of simulated station C;
- with the pattern continuing for other stations. A simulated ground station is normally named G, with address "add_G".

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

The following functions:

- lat:= CPR_LAT(y);
- lon:= CPR_LON(x);

are used to indicate that the given position will need to be encoded using the CPR encoding algorithm, currently defined in the ICAO VDL4 Technical Manual [1].

For example, the encoded position of the test station (station A) is:

- lat:= CPR_LAT(0);
- lon:= CPR_LON(0);

while the encoded position of a simulated station B, that is 325 NM away from A, will be expressed as:

- lat:= CPR_LAT(0);
- lon:= CPR_LON(E 325 NM).

The encoded offsets are written in the test scripts as follows:

- lat4:= CPR_LAT4(y);
- lon4:= CPR_LON4(x);

- lat6:= CPR_LAT6(y);
- lon6:= CPR_LON6(x);
- lat8:= CPR_LAT8(y);
- lon8:= CPR_LON8(x).

7.4.3.1.3 Tables of values for use in CPR test cases

The following tables are used in the CPR test cases.

Table 7.3: Key to CPR encoding table in following clause

| Table heading | Description |
|---------------|---|
| latitude | latitude to be encoded |
| longitude | longitude to be encoded |
| cpr type | CPR type of position report |
| lat enc | encoded latitude for transmission in fixed part of sync burst |
| lon enc | encoded longitude for transmission in fixed part of sync burst |
| pid enc | encoded patch id for transmission in variable part of sync burst |
| lat4(1-3) | encoded 4-bit high resolution latitude offset for transmission in variable part of sync burst |
| lat4(4) | encoded 4-bit high resolution latitude offset sign for transmission in variable part of sync burst |
| lat6(1-5) | encoded 6-bit high resolution latitude offset for transmission in variable part of sync burst |
| lat6(6) | encoded 6-bit high resolution latitude offset sign for transmission in variable part of sync burst |
| lat8(1-7) | encoded 8-bit high resolution latitude offset for transmission in variable part of sync burst |
| lat8(8) | encoded 8-bit high resolution latitude offset sign for transmission in variable part of sync burst |
| lat4(1-3) | encoded 4-bit high resolution longitude offset for transmission in variable part of sync burst |
| lat4(4) | encoded 4-bit high resolution longitude offset sign for transmission in variable part of sync burst |
| lat6(1-5) | encoded 6-bit high resolution longitude offset for transmission in variable part of sync burst |
| lat6(6) | encoded 6-bit high resolution longitude offset sign for transmission in variable part of sync burst |
| lat8(1-7) | encoded 8-bit high resolution longitude offset for transmission in variable part of sync burst |
| lat8(8) | encoded 8-bit high resolution longitude offset sign for transmission in variable part of sync burst |

7.4.3.1.3.1 Test values for CPR encoding CPR_ENC_TABLE (row, column) (CE(r, c))

(For the key to this table see the clause 7.4.3.1.3.)

The CPR test values have been designed assuming an aircraft travelling at constant velocity in a north-easterly direction and transmitting its position with a CPR report every 10 seconds. The receiving station is assumed to miss many of the transmitted reports, and in a way which allows this test to pass through all the various operations of the state machine during the decoding process (see table in clause 5.2.4.3.2.7). The missed positions, which are not relevant here, are excluded from the encoding table below and from the decoding table in clause 5.2.4.3.2.7 for clarity.

The input latitude and longitude values in the first two columns of the encoding table 7.4 (and in the first four columns of the table in clause 5.2.4.3.2.7) vary for the purpose of the test up to the fourth decimal place, but are required to be accurate to 9 decimal places as shown, in order to achieve the given encoded values.

Table 7.4: Table of test values for CPR position report encoding CPR_ENC_TABLE (row, column) (CE(r, c))

| latitude | longitude | cpr_type | lat_enc | lon_enc | pid_enc | lat4(1-3) | lat4(4) | lat6(1-5) | lat6(6) | lat8(1-7) | lat8(8) | lon4(1-3) | lon4(4) | lon6(1-5) | lon6(6) | lon8(1-7) | lon8(8) |
|--------------|--------------|----------|---------|---------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|
| 12,855700000 | -0,815000000 | 0 | 1169 | 15085 | 70 | 6 | 1 | 25 | 1 | 104 | 1 | 2 | 0 | 8 | 0 | 32 | 0 |
| 12,872000000 | -0,798700000 | 1 | 1030 | 15147 | 69 | 5 | 0 | 21 | 0 | 85 | 0 | 3 | 1 | 11 | 1 | 47 | 1 |
| 12,888300000 | -0,782400000 | 0 | 1183 | 15137 | 70 | 3 | 0 | 15 | 0 | 61 | 0 | 3 | 0 | 12 | 0 | 51 | 0 |
| 12,904600000 | -0,766100000 | 1 | 1043 | 15198 | 69 | 5 | 0 | 22 | 0 | 90 | 0 | 5 | 0 | 23 | 0 | 95 | 0 |
| 12,920900000 | -0,749800000 | 0 | 1196 | 15189 | 70 | 2 | 1 | 7 | 1 | 28 | 1 | 4 | 0 | 17 | 0 | 70 | 0 |
| 12,953500000 | -0,717200000 | 0 | 1209 | 15241 | 70 | 6 | 1 | 28 | 1 | 116 | 1 | 5 | 0 | 22 | 0 | 89 | 0 |
| 12,969800000 | -0,700900000 | 1 | 1069 | 15299 | 69 | 6 | 0 | 25 | 0 | 101 | 0 | 7 | 0 | 30 | 0 | 125 | 0 |
| 13,002400000 | -0,668300000 | 1 | 1082 | 15349 | 69 | 6 | 0 | 26 | 0 | 107 | 0 | 1 | 0 | 3 | 0 | 13 | 0 |
| 13,051300000 | -0,619400000 | 0 | 1250 | 15396 | 70 | 7 | 0 | 31 | 0 | 125 | 0 | 6 | 1 | 26 | 1 | 108 | 1 |
| 13,100200000 | -0,570500000 | 1 | 1121 | 15500 | 69 | 7 | 0 | 30 | 0 | 123 | 0 | 4 | 1 | 17 | 1 | 70 | 1 |
| 13,165400000 | -0,505300000 | 1 | 1146 | 15601 | 69 | 7 | 1 | 29 | 1 | 121 | 1 | 2 | 1 | 10 | 1 | 40 | 1 |
| 13,279500000 | -0,391200000 | 0 | 1343 | 15760 | 70 | 1 | 0 | 3 | 0 | 11 | 0 | 1 | 0 | 6 | 0 | 25 | 0 |
| 13,312100000 | -0,358600000 | 0 | 1356 | 15812 | 70 | 4 | 1 | 19 | 1 | 77 | 1 | 2 | 0 | 11 | 0 | 44 | 0 |
| 13,328400000 | -0,342300000 | 1 | 1211 | 15853 | 69 | 5 | 1 | 23 | 1 | 94 | 1 | 5 | 1 | 23 | 1 | 93 | 1 |
| 13,409900000 | -0,260800000 | 0 | 1396 | 15968 | 70 | 5 | 1 | 22 | 1 | 90 | 1 | 6 | 0 | 25 | 0 | 102 | 0 |
| 13,426200000 | -0,244500000 | 1 | 1250 | 16005 | 69 | 4 | 1 | 19 | 1 | 78 | 1 | 4 | 0 | 19 | 0 | 79 | 0 |
| 13,442500000 | -0,228200000 | 0 | 1410 | 16020 | 70 | 4 | 0 | 18 | 0 | 75 | 0 | 7 | 0 | 29 | 0 | 121 | 0 |
| 13,475100000 | -0,195600000 | 0 | 1423 | 16071 | 70 | 1 | 1 | 3 | 1 | 14 | 1 | 6 | 1 | 28 | 1 | 114 | 1 |
| 13,491400000 | -0,179300000 | 1 | 1276 | 16106 | 69 | 4 | 1 | 16 | 1 | 67 | 1 | 6 | 0 | 27 | 0 | 109 | 0 |
| 13,507700000 | -0,163000000 | 0 | 1436 | 16123 | 70 | 6 | 1 | 25 | 1 | 102 | 1 | 5 | 1 | 23 | 1 | 95 | 1 |
| 13,524000000 | -0,146700000 | 1 | 1289 | 16163 | 68 | 3 | 1 | 15 | 1 | 62 | 1 | 4 | 0 | 19 | 0 | 79 | 0 |
| 13,540300000 | -0,130400000 | 0 | 1450 | 16181 | 69 | 3 | 0 | 15 | 0 | 63 | 0 | 3 | 1 | 15 | 1 | 60 | 1 |
| 13,556600000 | -0,114100000 | 1 | 1302 | 16212 | 68 | 3 | 1 | 14 | 1 | 56 | 1 | 5 | 0 | 22 | 0 | 90 | 0 |
| 13,589200000 | -0,081500000 | 1 | 1315 | 16261 | 68 | 3 | 1 | 12 | 1 | 51 | 1 | 6 | 0 | 24 | 0 | 100 | 0 |
| 13,654400000 | -0,016300000 | 1 | 1341 | 16359 | 68 | 2 | 1 | 10 | 1 | 40 | 1 | 7 | 0 | 30 | 0 | 122 | 0 |
| 13,735900000 | 0,065200000 | 0 | 1530 | 101 | 36 | 2 | 0 | 9 | 0 | 38 | 0 | 2 | 0 | 7 | 0 | 30 | 0 |
| 13,850000000 | 0,179300000 | 1 | 1419 | 269 | 36 | 0 | 1 | 2 | 1 | 8 | 1 | 4 | 1 | 17 | 1 | 68 | 1 |
| 13,866300000 | 0,195600000 | 0 | 1583 | 303 | 36 | 3 | 1 | 15 | 1 | 63 | 1 | 5 | 0 | 22 | 0 | 89 | 0 |
| 13,996700000 | 0,326000000 | 0 | 1637 | 504 | 36 | 5 | 0 | 22 | 0 | 89 | 0 | 6 | 1 | 26 | 1 | 105 | 1 |
| 14,013000000 | 0,342300000 | 1 | 1484 | 514 | 36 | 1 | 0 | 5 | 0 | 19 | 0 | 1 | 1 | 4 | 1 | 15 | 1 |
| 14,143400000 | 0,472700000 | 1 | 1536 | 710 | 36 | 2 | 0 | 10 | 0 | 40 | 0 | 2 | 0 | 7 | 0 | 28 | 0 |
| 14,208600000 | 0,537900000 | 1 | 1562 | 808 | 36 | 3 | 0 | 12 | 0 | 51 | 0 | 3 | 0 | 12 | 0 | 50 | 0 |
| 14,290100000 | 0,619400000 | 0 | 1757 | 958 | 36 | 3 | 0 | 13 | 0 | 52 | 0 | 5 | 1 | 24 | 1 | 98 | 1 |

| latitude | longitude | cpr_type | lat_enc | lon_enc | pid_enc | lat4(1-3) | lat4(4) | lat6(1-5) | lat6(6) | lat8(1-7) | lat8(8) | lon4(1-3) | lon4(4) | lon6(1-5) | lon6(6) | lon8(1-7) | lon8(8) |
|--------------|--------------|----------|---------|---------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|
| 14,322700000 | 0,652000000 | 0 | 1770 | 1009 | 36 | 2 | 1 | 9 | 1 | 37 | 1 | 2 | 0 | 11 | 0 | 44 | 0 |
| 14,404200000 | 0,733500000 | 1 | 1640 | 1102 | 36 | 5 | 0 | 20 | 0 | 83 | 0 | 6 | 0 | 28 | 0 | 114 | 0 |
| 14,436800000 | 0,766100000 | 1 | 1653 | 1151 | 36 | 5 | 0 | 22 | 0 | 89 | 0 | 7 | 0 | 30 | 0 | 125 | 0 |
| 14,518300000 | 0,847600000 | 0 | 1850 | 1311 | 36 | 3 | 1 | 15 | 1 | 62 | 1 | 7 | 1 | 30 | 1 | 121 | 1 |
| 14,550900000 | 0,880200000 | 0 | 1864 | 1362 | 36 | 6 | 0 | 25 | 0 | 103 | 0 | 1 | 0 | 5 | 0 | 21 | 0 |
| 14,632400000 | 0,961700000 | 1 | 1731 | 1444 | 36 | 7 | 0 | 29 | 0 | 121 | 0 | 4 | 1 | 16 | 1 | 65 | 1 |
| 14,713900000 | 1,043200000 | 0 | 1930 | 1614 | 36 | 5 | 1 | 21 | 1 | 87 | 1 | 2 | 1 | 8 | 1 | 32 | 1 |
| 14,746500000 | 1,075800000 | 0 | 1944 | 1665 | 36 | 4 | 0 | 19 | 0 | 78 | 0 | 6 | 0 | 27 | 0 | 110 | 0 |
| 14,762800000 | 1,092100000 | 1 | 1782 | 1640 | 36 | 6 | 1 | 27 | 1 | 112 | 1 | 1 | 1 | 5 | 1 | 22 | 1 |
| 14,876900000 | 1,206200000 | 0 | 1997 | 1866 | 36 | 1 | 1 | 6 | 1 | 23 | 1 | 5 | 1 | 21 | 1 | 85 | 1 |
| 14,893200000 | 1,222500000 | 1 | 1834 | 1836 | 36 | 5 | 1 | 22 | 1 | 90 | 1 | 1 | 0 | 5 | 0 | 20 | 0 |
| 15,007300000 | 1,336600000 | 0 | 2050 | 2068 | 36 | 7 | 1 | 30 | 1 | 124 | 1 | 1 | 1 | 6 | 1 | 25 | 1 |
| 15,088800000 | 1,418100000 | 1 | 1912 | 2130 | 36 | 3 | 1 | 14 | 1 | 58 | 1 | 5 | 0 | 21 | 0 | 85 | 0 |
| 15,121400000 | 1,450700000 | 1 | 1925 | 2179 | 36 | 3 | 1 | 13 | 1 | 53 | 1 | 5 | 0 | 23 | 0 | 95 | 0 |
| 15,154000000 | 1,483300000 | 1 | 1938 | 2228 | 36 | 3 | 1 | 12 | 1 | 47 | 1 | 6 | 0 | 26 | 0 | 106 | 0 |
| 15,235500000 | 1,564800000 | 0 | 2144 | 2421 | 36 | 1 | 0 | 4 | 0 | 16 | 0 | 3 | 1 | 12 | 1 | 48 | 1 |
| 15,268100000 | 1,597400000 | 0 | 2157 | 2472 | 36 | 4 | 1 | 18 | 1 | 73 | 1 | 5 | 0 | 23 | 0 | 94 | 0 |
| 15,349600000 | 1,678900000 | 1 | 2016 | 2521 | 36 | 1 | 1 | 4 | 1 | 15 | 1 | 5 | 1 | 20 | 1 | 84 | 1 |
| 15,365900000 | 1,695200000 | 0 | 2197 | 2623 | 36 | 5 | 1 | 21 | 1 | 85 | 1 | 1 | 0 | 3 | 0 | 12 | 0 |
| 15,480000000 | 1,809300000 | 1 | 2068 | 2717 | 36 | 0 | 0 | 2 | 0 | 6 | 0 | 2 | 1 | 10 | 1 | 41 | 1 |
| 15,496300000 | 1,825600000 | 0 | 2251 | 2825 | 36 | 4 | 0 | 16 | 0 | 67 | 0 | 4 | 0 | 17 | 0 | 71 | 0 |
| 15,610400000 | 1,939700000 | 1 | 2120 | 2913 | 36 | 2 | 0 | 7 | 0 | 28 | 0 | 0 | 0 | 0 | 0 | 2 | 0 |
| 15,626700000 | 1,956000000 | 0 | 2304 | 3026 | 36 | 2 | 1 | 8 | 1 | 34 | 1 | 7 | 1 | 30 | 1 | 123 | 1 |
| 15,740800000 | 2,070100000 | 1 | 2172 | 3109 | 36 | 3 | 0 | 12 | 0 | 49 | 0 | 2 | 0 | 11 | 0 | 45 | 0 |
| 15,757100000 | 2,086400000 | 0 | 2358 | 3228 | 36 | 7 | 0 | 29 | 0 | 119 | 0 | 4 | 1 | 16 | 1 | 64 | 1 |
| 15,871200000 | 2,200500000 | 1 | 2224 | 3305 | 36 | 4 | 0 | 17 | 0 | 71 | 0 | 5 | 0 | 21 | 0 | 87 | 0 |
| 15,887500000 | 2,216800000 | 0 | 2411 | 3430 | 36 | 1 | 0 | 4 | 0 | 17 | 0 | 0 | 1 | 1 | 1 | 4 | 1 |
| 15,903800000 | 2,233100000 | 1 | 2237 | 3354 | 36 | 4 | 0 | 19 | 0 | 76 | 0 | 5 | 0 | 24 | 0 | 98 | 0 |
| 15,920100000 | 2,249400000 | 0 | 2424 | 3480 | 36 | 4 | 1 | 17 | 1 | 71 | 1 | 6 | 1 | 28 | 1 | 117 | 1 |
| | | | | | | | | | | | | | | | | | |
| 19,098600000 | 5,427900000 | 1 | 3509 | 8151 | 36 | 5 | 0 | 23 | 0 | 94 | 0 | 7 | 1 | 30 | 1 | 123 | 1 |
| 19,114900000 | 5,444200000 | 0 | 3733 | 8424 | 36 | 6 | 0 | 28 | 0 | 114 | 0 | 4 | 0 | 17 | 0 | 71 | 0 |
| 19,131200000 | 5,460500000 | 1 | 3522 | 8200 | 36 | 5 | 0 | 24 | 0 | 99 | 0 | 6 | 1 | 27 | 1 | 112 | 1 |
| 19,147500000 | 5,476800000 | 0 | 3746 | 8474 | 36 | 1 | 0 | 6 | 0 | 25 | 0 | 2 | 1 | 10 | 1 | 41 | 1 |
| 19,163800000 | 5,493100000 | 1 | 3535 | 7999 | 36 | 6 | 0 | 26 | 0 | 105 | 0 | 6 | 1 | 26 | 1 | 106 | 1 |
| 19,180100000 | 5,509400000 | 0 | 3759 | 8274 | 36 | 4 | 1 | 16 | 1 | 64 | 1 | 2 | 0 | 7 | 0 | 31 | 0 |
| | | | | | | | | | | | | | | | | | |
| 23,483300000 | 9,812600000 | 1 | 1159 | 14290 | 72 | 4 | 1 | 18 | 1 | 73 | 1 | 3 | 0 | 15 | 0 | 60 | 0 |
| 23,499600000 | 9,828900000 | 0 | 1433 | 14761 | 72 | 1 | 1 | 5 | 1 | 22 | 1 | 3 | 0 | 13 | 0 | 52 | 0 |
| 23,515900000 | 9,845200000 | 1 | 1172 | 14337 | 72 | 4 | 1 | 17 | 1 | 68 | 1 | 3 | 1 | 15 | 1 | 60 | 1 |
| 23,532200000 | 9,861500000 | 0 | 1446 | 14361 | 72 | 6 | 1 | 27 | 1 | 111 | 1 | 0 | 0 | 2 | 0 | 7 | 0 |
| 23,548500000 | 9,877800000 | 1 | 1185 | 13935 | 72 | 3 | 1 | 15 | 1 | 63 | 1 | 3 | 1 | 12 | 1 | 48 | 1 |
| | | | | | | | | | | | | | | | | | |
| 27,167100000 | 13,496400000 | 0 | 2935 | 3271 | 73 | 1 | 0 | 4 | 0 | 18 | 0 | 5 | 1 | 22 | 1 | 91 | 1 |
| 27,183400000 | 13,512700000 | 1 | 2632 | 2680 | 73 | 5 | 1 | 24 | 1 | 99 | 1 | 2 | 1 | 10 | 1 | 39 | 1 |
| 27,199700000 | 13,529000000 | 0 | 2948 | 3319 | 73 | 4 | 1 | 17 | 1 | 70 | 1 | 2 | 0 | 10 | 0 | 43 | 0 |
| 27,216000000 | 13,545300000 | 1 | 2645 | 2726 | 73 | 5 | 1 | 23 | 1 | 94 | 1 | 2 | 1 | 9 | 1 | 37 | 1 |
| 27,232300000 | 13,561600000 | 0 | 2962 | 2749 | 73 | 5 | 0 | 23 | 0 | 95 | 0 | 2 | 1 | 9 | 1 | 36 | 1 |
| 27,248600000 | 13,577900000 | 1 | 2658 | 2154 | 73 | 5 | 1 | 22 | 1 | 89 | 1 | 3 | 1 | 14 | 1 | 58 | 1 |
| | | | | | | | | | | | | | | | | | |

| latitude | longitude | cpr_type | lat_enc | lon_enc | pid_enc | lat4(1-3) | lat4(4) | lat6(1-5) | lat6(6) | lat8(1-7) | lat8(8) | lon4(1-3) | lon4(4) | lon6(1-5) | lon6(6) | lon8(1-7) | lon8(8) |
|--------------|--------------|----------|---------|---------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|
| 30,361900000 | 16,691200000 | 0 | 148 | 7164 | 109 | 3 | 1 | 12 | 1 | 50 | 1 | 3 | 1 | 15 | 1 | 63 | 1 |
| 30,378200000 | 16,707500000 | 1 | 3904 | 6427 | 73 | 4 | 1 | 20 | 1 | 82 | 1 | 1 | 0 | 5 | 0 | 22 | 0 |
| 30,394500000 | 16,723800000 | 0 | 162 | 7210 | 109 | 6 | 0 | 28 | 0 | 115 | 0 | 3 | 1 | 15 | 1 | 61 | 1 |
| 30,508600000 | 16,837900000 | 1 | 3956 | 6605 | 73 | 3 | 1 | 15 | 1 | 60 | 1 | 1 | 0 | 4 | 0 | 14 | 0 |
| 30,524900000 | 16,854200000 | 0 | 215 | 6627 | 109 | 1 | 0 | 3 | 0 | 14 | 0 | 3 | 1 | 12 | 1 | 50 | 1 |
| 30,541200000 | 16,870500000 | 1 | 3969 | 5882 | 73 | 3 | 1 | 13 | 1 | 55 | 1 | 4 | 0 | 18 | 0 | 76 | 0 |
| 30,557500000 | 16,886800000 | 0 | 228 | 6672 | 109 | 4 | 1 | 18 | 1 | 75 | 1 | 4 | 0 | 18 | 0 | 75 | 0 |
| | | | | | | | | | | | | | | | | | |
| 33,361100000 | 19,690400000 | 0 | 1376 | 10499 | 109 | 5 | 1 | 23 | 1 | 94 | 1 | 4 | 1 | 20 | 1 | 81 | 1 |
| 33,377400000 | 19,706700000 | 1 | 1003 | 9625 | 109 | 5 | 1 | 23 | 1 | 96 | 1 | 3 | 0 | 15 | 0 | 63 | 0 |
| 33,393700000 | 19,723000000 | 0 | 1390 | 10544 | 109 | 4 | 0 | 17 | 0 | 71 | 0 | 2 | 0 | 11 | 0 | 44 | 0 |
| 33,507800000 | 19,837100000 | 1 | 1055 | 9797 | 109 | 4 | 1 | 18 | 1 | 74 | 1 | 2 | 0 | 9 | 0 | 39 | 0 |
| 33,524100000 | 19,853400000 | 0 | 1443 | 9818 | 109 | 2 | 1 | 7 | 1 | 30 | 1 | 5 | 1 | 22 | 1 | 91 | 1 |
| 33,540400000 | 19,869700000 | 1 | 1068 | 8936 | 109 | 4 | 1 | 17 | 1 | 69 | 1 | 5 | 0 | 23 | 0 | 93 | 0 |
| 33,556700000 | 19,886000000 | 0 | 1456 | 9861 | 109 | 7 | 1 | 29 | 1 | 119 | 1 | 5 | 1 | 24 | 1 | 97 | 1 |
| | | | | | | | | | | | | | | | | | |
| 36,132100000 | 22,461400000 | 1 | 2100 | 12238 | 109 | 1 | 1 | 6 | 1 | 23 | 1 | 1 | 1 | 4 | 1 | 16 | 1 |
| 36,148400000 | 22,477700000 | 0 | 2518 | 13282 | 109 | 3 | 0 | 14 | 0 | 58 | 0 | 3 | 0 | 15 | 0 | 62 | 0 |
| 36,164700000 | 22,494000000 | 1 | 2113 | 12280 | 109 | 1 | 1 | 4 | 1 | 18 | 1 | 6 | 0 | 25 | 0 | 100 | 0 |
| 36,278800000 | 22,608100000 | 0 | 2571 | 13454 | 109 | 2 | 1 | 10 | 1 | 43 | 1 | 2 | 0 | 9 | 0 | 38 | 0 |
| 36,295100000 | 22,624400000 | 1 | 2165 | 11416 | 109 | 0 | 0 | 1 | 0 | 3 | 0 | 2 | 1 | 10 | 1 | 42 | 1 |
| 36,311400000 | 22,640700000 | 0 | 2585 | 12467 | 109 | 7 | 0 | 30 | 0 | 122 | 0 | 7 | 0 | 29 | 0 | 118 | 0 |
| 36,327700000 | 22,657000000 | 1 | 2178 | 11456 | 109 | 0 | 0 | 2 | 0 | 9 | 0 | 3 | 1 | 14 | 1 | 56 | 1 |
| | | | | | | | | | | | | | | | | | |
| 38,723800000 | 25,053100000 | 1 | 3132 | 14400 | 109 | 1 | 0 | 5 | 0 | 22 | 0 | 5 | 1 | 23 | 1 | 94 | 1 |
| 38,740100000 | 25,069400000 | 0 | 3579 | 15561 | 109 | 1 | 1 | 4 | 1 | 18 | 1 | 4 | 1 | 16 | 1 | 67 | 1 |
| 38,756400000 | 25,085700000 | 1 | 3145 | 14440 | 109 | 2 | 0 | 7 | 0 | 27 | 0 | 6 | 1 | 26 | 1 | 108 | 1 |
| 38,870500000 | 25,199800000 | 0 | 3632 | 15727 | 109 | 7 | 1 | 29 | 1 | 119 | 1 | 6 | 1 | 26 | 1 | 108 | 1 |
| 38,886800000 | 25,216100000 | 1 | 3197 | 13453 | 109 | 3 | 0 | 12 | 0 | 49 | 0 | 2 | 1 | 7 | 1 | 28 | 1 |
| 38,903100000 | 25,232400000 | 0 | 3646 | 14621 | 109 | 3 | 0 | 11 | 0 | 46 | 0 | 4 | 0 | 20 | 0 | 81 | 0 |
| 38,919400000 | 25,248700000 | 1 | 3210 | 13492 | 109 | 3 | 0 | 13 | 0 | 54 | 0 | 4 | 0 | 20 | 0 | 81 | 0 |
| | | | | | | | | | | | | | | | | | |
| 41,185100000 | 27,514400000 | 0 | 485 | 1042 | 146 | 4 | 1 | 19 | 1 | 76 | 1 | 5 | 0 | 23 | 0 | 94 | 0 |
| 41,201400000 | 27,530700000 | 1 | 23 | 16192 | 145 | 4 | 1 | 19 | 1 | 78 | 1 | 3 | 0 | 13 | 0 | 55 | 0 |
| 41,217700000 | 27,547000000 | 0 | 499 | 1082 | 146 | 5 | 0 | 22 | 0 | 89 | 0 | 4 | 0 | 19 | 0 | 79 | 0 |
| 41,331800000 | 27,661100000 | 1 | 75 | 16346 | 145 | 3 | 1 | 14 | 1 | 57 | 1 | 1 | 1 | 5 | 1 | 19 | 1 |
| 41,348100000 | 27,677400000 | 0 | 552 | 16365 | 145 | 1 | 1 | 3 | 1 | 12 | 1 | 5 | 1 | 22 | 1 | 92 | 1 |
| 41,364400000 | 27,693700000 | 1 | 88 | 15124 | 145 | 3 | 1 | 13 | 1 | 51 | 1 | 5 | 1 | 22 | 1 | 90 | 1 |
| 41,380700000 | 27,710000000 | 0 | 565 | 21 | 146 | 6 | 1 | 25 | 1 | 101 | 1 | 1 | 0 | 4 | 0 | 17 | 0 |
| | | | | | | | | | | | | | | | | | |
| 43,532300000 | 29,861600000 | 0 | 1446 | 2567 | 146 | 7 | 1 | 30 | 1 | 121 | 1 | 4 | 0 | 16 | 0 | 65 | 0 |
| 43,548600000 | 29,877900000 | 1 | 958 | 1226 | 146 | 3 | 0 | 13 | 0 | 54 | 0 | 5 | 1 | 21 | 1 | 85 | 1 |
| 43,564900000 | 29,894200000 | 0 | 1460 | 2605 | 146 | 2 | 0 | 11 | 0 | 44 | 0 | 4 | 1 | 20 | 1 | 80 | 1 |
| 43,679000000 | 30,008300000 | 1 | 1010 | 1375 | 146 | 4 | 0 | 18 | 0 | 76 | 0 | 4 | 0 | 19 | 0 | 78 | 0 |
| 43,695300000 | 30,024600000 | 0 | 1513 | 1393 | 146 | 3 | 1 | 14 | 1 | 57 | 1 | 3 | 1 | 15 | 1 | 60 | 1 |
| 43,711600000 | 30,040900000 | 1 | 1023 | 45 | 146 | 4 | 0 | 20 | 0 | 81 | 0 | 5 | 0 | 20 | 0 | 84 | 0 |
| 43,727900000 | 30,057200000 | 0 | 1527 | 1430 | 146 | 6 | 0 | 26 | 0 | 108 | 0 | 5 | 1 | 20 | 1 | 83 | 1 |
| | | | | | | | | | | | | | | | | | |
| 45,781700000 | 32,111000000 | 1 | 1847 | 2306 | 146 | 2 | 0 | 10 | 0 | 41 | 0 | 5 | 0 | 23 | 0 | 93 | 0 |
| 45,798000000 | 32,127300000 | 0 | 2374 | 3785 | 146 | 4 | 1 | 17 | 1 | 71 | 1 | 7 | 1 | 31 | 1 | 126 | 1 |

| latitude | longitude | cpr_type | lat_enc | lon_enc | pid_enc | lat4(1-3) | lat4(4) | lat6(1-5) | lat6(6) | lat8(1-7) | lat8(8) | lon4(1-3) | lon4(4) | lon6(1-5) | lon6(6) | lon8(1-7) | lon8(8) |
|--------------|--------------|----------|---------|---------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|-----------|---------|
| 45,814300000 | 32,143600000 | 1 | 1860 | 2341 | 146 | 3 | 0 | 11 | 0 | 46 | 0 | 3 | 1 | 15 | 1 | 61 | 1 |
| 45,928400000 | 32,257700000 | 0 | 2428 | 3934 | 146 | 4 | 0 | 20 | 0 | 81 | 0 | 2 | 0 | 9 | 0 | 37 | 0 |
| 45,944700000 | 32,274000000 | 1 | 1912 | 1015 | 146 | 4 | 0 | 17 | 0 | 68 | 0 | 1 | 0 | 5 | 0 | 19 | 0 |
| 45,961000000 | 32,290300000 | 0 | 2441 | 2501 | 146 | 0 | 1 | 2 | 1 | 7 | 1 | 7 | 1 | 29 | 1 | 118 | 1 |
| 45,977300000 | 32,306600000 | 1 | 1925 | 1049 | 146 | 4 | 0 | 18 | 0 | 73 | 0 | 1 | 1 | 3 | 1 | 12 | 1 |
| 47,949600000 | 34,278900000 | 1 | 2710 | 3113 | 146 | 1 | 0 | 4 | 0 | 17 | 0 | 6 | 1 | 27 | 1 | 111 | 1 |
| 47,965900000 | 34,295200000 | 0 | 3262 | 4691 | 146 | 1 | 1 | 2 | 1 | 9 | 1 | 3 | 1 | 13 | 1 | 55 | 1 |
| 47,982200000 | 34,311500000 | 1 | 2723 | 3148 | 146 | 1 | 0 | 5 | 0 | 22 | 0 | 6 | 0 | 27 | 0 | 111 | 0 |
| 48,096300000 | 34,425600000 | 0 | 3315 | 4834 | 146 | 6 | 1 | 27 | 1 | 110 | 1 | 5 | 0 | 22 | 0 | 91 | 0 |
| 48,112600000 | 34,441900000 | 1 | 2775 | 1717 | 146 | 2 | 0 | 11 | 0 | 43 | 0 | 5 | 0 | 21 | 0 | 87 | 0 |
| 48,128900000 | 34,458200000 | 0 | 3329 | 3301 | 146 | 3 | 0 | 13 | 0 | 55 | 0 | 2 | 1 | 7 | 1 | 28 | 1 |
| 48,145200000 | 34,474500000 | 1 | 2788 | 1749 | 146 | 3 | 0 | 12 | 0 | 49 | 0 | 4 | 1 | 18 | 1 | 75 | 1 |

7.4.3.1.3.2 CPR test value tolerances

The number of decimal places afforded to the decoded latitude and longitude values in the last eight columns of the table in clause 5.2.4.3.2.7 varies according to the resolution expected from the decoding algorithm (see table 7.5). When using the table in clause 5.2.4.3.2.7 to test the validity of an installed algorithm, the given decoded latitude and longitude values should be interpreted using the tolerances given in the last column of table 7.2.

Table 7.5: CPR test value tolerances

| Decoded Parameter | Total number of bits used to encode | Approximate max decoded error (degrees) | Number of decimal places given to decoded values | Tolerance to be given on decoded values during test of algorithm (degrees) |
|---|-------------------------------------|---|--|--|
| Decoded lat | 12 | $\pm 0,0012$ | 4 | $\pm 0,0003$ |
| Decoded lon | 14 | $\pm 0,0012$ (see note 1) | 5 | $\pm 0,0003$ |
| Decoded 4-bit lat | 16 | $\pm 0,000076$ | 6 | $\pm 0,00002$ |
| Decoded 4-bit lon | 18 | $\pm 0,000076$ (see note 1) | 6 | $\pm 0,00002$ |
| Decoded 6-bit lat | 18 | $\pm 0,000019$ | 6 | $\pm 0,000005$ |
| Decoded 6-bit lon | 20 | $\pm 0,000019$ (see note 1) | 7 | $\pm 0,000005$ |
| Decoded 8-bit lat | 20 | $\pm 0,000048$ | 7 | $\pm 0,000012$ |
| Decoded 8-bit lon | 22 | $\pm 0,000048$ (see note 1) | 7 | $\pm 0,000012$ |
| NOTE 1: Take into account that in the case of longitude the maximum error in the decoded value is up to four times greater at high latitudes. | | | | |
| NOTE 2: Since the figures given in the table in clause 7.4.3.1.5 are designed to be used to test the CPR algorithm, the decoded figures in the last eight columns of the table are given to more decimal places than the number of decimal places to which the decoded results may be relied upon for position reporting. | | | | |

7.4.3.1.3.3 Key to CPR decoding table in following clause

Table 7.6: Key to CPR decoding table in following clause

| Table heading | Description |
|----------------------|---|
| lat last | input latitude from last position |
| lat prev | input latitude from previous (last but one) position |
| lon last | input longitude from last position |
| lon prev | input longitude from previous (last but one) position |
| tl | input CPR type from last position |
| tp | input CPR type from previous (last but one) position |
| tr | indicates 1 if the two points straddle a transition latitude (and 0 otherwise) |
| i | initial state in CPR state machine |
| p | indicates 1 if a patch ID is available for decoding (and 0 otherwise) |
| tim | time in seconds since last report received |
| o | indicates whether the receiver knows its own position |
| op | state machine operation used (see Op field in tables in ED-108 Part 1 Appendix C) |
| cal | calculation determined by state machine operation |
| f | final state in CPR state machine |
| decoded lat | decoded latitude without high resolution offset |
| decoded lon | decoded longitude without high resolution offset |
| decoded 4-bit lat | decoded latitude with 4-bit high resolution offset |
| decoded 4-bit lon | decoded longitude with 4-bit high resolution offset |
| decoded 6-bit lat | decoded latitude with 6-bit high resolution offset |
| decoded 6-bit lon | decoded longitude with 6-bit high resolution offset |
| decoded 8-bit lat | decoded latitude with 8-bit high resolution offset |
| decoded 8-bit lon | decoded longitude with 8-bit high resolution offset |

7.4.3.1.3.4 Test values for CPR decoding CPR_DEC_TABLE (row, column) (CD(r, c))

(For the key to this table see the clause 7.4.3.1.3.3.)

The encoding for the latitude and longitude values contained in the first four columns of the decoding table 7.7 is given in the encoding table in clause 5.2.4.3.2.4.

The decoding operation used in table 7.7 is determined by the state machine tables in clauses 5.2.4.3.2.1 and 5.2.4.3.2.2, and referred to in this table using the column with heading "op". The time since a report was last received is indicated by the column headed "tim". When plotted, the decoded positions form a straight line in a north-easterly direction (allowing for the expected decoding errors), with gaps in the line corresponding to the missed reports.

Table 7.7: Table of test values for CPR position report decoding CPR_DEC_TABLE (row, column) (CD(r, c))

| lat last | lat prev | lon last | lon prev | tl | tp | tr | i | p | tim | o | op | cal | f | decoded lat | decoded lon | decoded 4-bit lat | decoded 4-bit lon | decoded 6-bit lat | decoded 6-bit lon | decoded 8-bit lat | decoded 8-bit lon |
|--------------|--------------|--------------|--------------|----|----|----|---|---|-----|---|----|-----|---|-------------|-------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| 12,855700000 | - | -0,815000000 | - | 0 | - | - | 1 | n | - | n | 1c | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 12,872000000 | 12,855700000 | -0,798700000 | -0,815000000 | 1 | 0 | 0 | 2 | n | 10 | n | 2k | GL | 4 | 12,8728 | -0,79882 | 12,871944 | -0,798681 | 12,871991 | -0,7987048 | 12,8720009 | -0,7986998 |
| 12,888300000 | 12,872000000 | -0,782400000 | -0,798700000 | 0 | 1 | 0 | 4 | n | 10 | n | 4i | GL | 4 | 12,8889 | -0,78227 | 12,888366 | -0,782409 | 12,888298 | -0,7823958 | 12,8883024 | -0,7824004 |
| 12,904600000 | 12,888300000 | -0,766100000 | -0,782400000 | 1 | 0 | 0 | 4 | n | 10 | n | 4q | GL | 4 | 12,9055 | -0,76586 | 12,904597 | -0,766089 | 12,904603 | -0,7660982 | 12,9046045 | -0,7661002 |
| 12,920900000 | 12,904600000 | -0,749800000 | -0,766100000 | 0 | 1 | 0 | 4 | n | 10 | n | 4i | GL | 4 | 12,9206 | -0,74963 | 12,920984 | -0,749807 | 12,920911 | -0,7497994 | 12,9209041 | -0,7498003 |
| 12,953500000 | 12,920900000 | -0,717200000 | -0,749800000 | 0 | 0 | 0 | 4 | n | 20 | n | 4g | L2 | 4 | 12,9524 | -0,71698 | 12,953428 | -0,717204 | 12,953484 | -0,7172029 | 12,9534962 | -0,7172001 |
| 12,969800000 | 12,953500000 | -0,700900000 | -0,717200000 | 1 | 0 | 0 | 4 | n | 10 | n | 4q | GL | 4 | 12,9708 | -0,70058 | 12,969724 | -0,700906 | 12,969788 | -0,7008955 | 12,9698019 | -0,7009008 |
| 13,002400000 | 12,969800000 | -0,668300000 | -0,700900000 | 1 | 1 | 0 | 4 | n | 20 | n | 4r | L2 | 4 | 13,0035 | -0,66827 | 13,002377 | -0,668314 | 13,002400 | -0,6682993 | 13,0023956 | -0,6683011 |
| 13,051300000 | 13,002400000 | -0,619400000 | -0,668300000 | 0 | 1 | 0 | 4 | n | 30 | n | 4i | GL | 4 | 13,0525 | -0,61967 | 13,051282 | -0,619398 | 13,051282 | -0,6194034 | 13,0513013 | -0,6193998 |
| 13,100200000 | 13,051300000 | -0,570500000 | -0,619400000 | 1 | 0 | 0 | 4 | n | 30 | n | 4q | GL | 4 | 13,1014 | -0,57068 | 13,100157 | -0,570493 | 13,100197 | -0,5705004 | 13,1001965 | -0,5704995 |
| 13,165400000 | 13,100200000 | -0,505300000 | -0,570500000 | 1 | 1 | 0 | 4 | n | 40 | n | 4t | L2 | 4 | 13,1642 | -0,50540 | 13,165463 | -0,505310 | 13,165382 | -0,5052977 | 13,1654038 | -0,5053002 |
| 13,279500000 | 13,165400000 | -0,391200000 | -0,505300000 | 0 | 1 | 0 | 4 | n | 70 | n | 4k | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 13,312100000 | 13,279500000 | -0,358600000 | -0,391200000 | 0 | 0 | 0 | 2 | n | 20 | n | 2e | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 13,328400000 | 13,312100000 | -0,342300000 | -0,358600000 | 1 | 0 | 0 | 2 | n | 10 | n | 2k | GL | 4 | 13,3275 | -0,34254 | 13,328370 | -0,342305 | 13,328404 | -0,3422961 | 13,3284021 | -0,3422992 |
| 13,409900000 | 13,328400000 | -0,260800000 | -0,342300000 | 0 | 1 | 0 | 4 | n | 50 | n | 4j | L2 | 4 | 13,4090 | -0,26055 | 13,409908 | -0,260818 | 13,409902 | -0,2608020 | 13,4099007 | -0,2608009 |
| 13,426200000 | 13,409900000 | -0,244500000 | -0,260800000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 13,4254 | -0,24430 | 13,426149 | -0,244484 | 13,426201 | -0,2444972 | 13,4262030 | -0,2445002 |
| 13,442500000 | 13,426200000 | -0,228200000 | -0,244500000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 13,4432 | -0,22790 | 13,442526 | -0,228216 | 13,442514 | -0,2281954 | 13,4425024 | -0,2282008 |
| 13,475100000 | 13,442500000 | -0,195600000 | -0,228200000 | 0 | 0 | 0 | 4 | n | 10 | y | 4b | L2 | 4 | 13,4750 | -0,19588 | 13,475144 | -0,195613 | 13,475088 | -0,1955990 | 13,4751041 | -0,1956007 |
| 13,491400000 | 13,475100000 | -0,179300000 | -0,195600000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 13,4907 | -0,17902 | 13,491455 | -0,179300 | 13,491386 | -0,1793049 | 13,4914004 | -0,1793008 |
| 13,507700000 | 13,491400000 | -0,163000000 | -0,179300000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 13,5067 | -0,16324 | 13,507762 | -0,163011 | 13,507700 | -0,1630025 | 13,5076962 | -0,1630006 |
| 13,524000000 | 13,507700000 | -0,146700000 | -0,163000000 | 1 | 0 | 1 | 4 | n | 10 | y | 4z | L2 | 4 | 13,5234 | -0,14649 | 13,523929 | -0,146684 | 13,523999 | -0,1466974 | 13,5240040 | -0,1467004 |
| 13,540300000 | 13,524000000 | -0,130400000 | -0,146700000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 13,5409 | -0,13055 | 13,540380 | -0,130413 | 13,540313 | -0,1303950 | 13,5402978 | -0,1303987 |
| 13,556600000 | 13,540300000 | -0,114100000 | -0,130400000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 13,5560 | -0,11387 | 13,556582 | -0,114103 | 13,556611 | -0,1141015 | 13,5565977 | -0,1141012 |
| 13,589200000 | 13,556600000 | -0,081500000 | -0,114100000 | 1 | 1 | 0 | 4 | n | 10 | y | 4m | L2 | 4 | 13,5887 | -0,08124 | 13,589235 | -0,081523 | 13,589183 | -0,0814950 | 13,5892013 | -0,0814994 |
| 13,654400000 | 13,589200000 | -0,016300000 | -0,081500000 | 1 | 1 | 0 | 4 | n | 40 | y | 4o | L2 | 4 | 13,6540 | -0,01598 | 13,654362 | -0,016314 | 13,654408 | -0,0163033 | 13,6543987 | -0,0163009 |
| 13,735900000 | 13,654400000 | 0,065200000 | -0,016300000 | 0 | 1 | 0 | 4 | n | 50 | y | 4e | L2 | 4 | 13,7363 | 0,06528 | 13,735915 | 0,065183 | 13,735909 | 0,0652027 | 13,7358984 | 0,0651994 |
| 13,850000000 | 13,735900000 | 0,179300000 | 0,065200000 | 1 | 0 | 0 | 4 | n | 70 | y | 4p | L1 | 3 | 13,8499 | 0,17912 | 13,849922 | 0,179312 | 13,850003 | 0,1793040 | 13,8500006 | 0,1792996 |
| 13,866300000 | 13,850000000 | 0,195600000 | 0,179300000 | 0 | 1 | 0 | 3 | n | 10 | y | 3c | GL | 4 | 13,8657 | 0,19583 | 13,866213 | 0,195596 | 13,866281 | 0,1955978 | 13,8662956 | 0,1956006 |
| 13,996700000 | 13,866300000 | 0,326000000 | 0,195600000 | 0 | 0 | 0 | 4 | n | 80 | y | 4f | L1 | 3 | 13,9976 | 0,32573 | 13,996686 | 0,326009 | 13,996691 | 0,3260032 | 13,9967023 | 0,3259994 |
| 14,013000000 | 13,996700000 | 0,342300000 | 0,326000000 | 1 | 0 | 0 | 3 | n | 10 | y | 3h | GL | 4 | 14,0132 | 0,34226 | 14,013007 | 0,342309 | 14,012984 | 0,3423046 | 14,0129989 | 0,3423010 |
| 14,143400000 | 14,013000000 | 0,472700000 | 0,342300000 | 1 | 1 | 0 | 4 | n | 80 | n | 4u | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 14,208600000 | 14,143400000 | 0,537900000 | 0,472700000 | 1 | 1 | 0 | 2 | n | 40 | n | 2m | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 14,290100000 | 14,208600000 | 0,619400000 | 0,537900000 | 0 | 1 | 0 | 2 | n | 50 | y | 2d | L1 | 3 | 14,2906 | 0,61915 | 14,290075 | 0,619381 | 14,290086 | 0,6193999 | 14,2900984 | 0,6193991 |
| 14,322700000 | 14,290100000 | 0,652000000 | 0,619400000 | 0 | 0 | 0 | 3 | n | 20 | y | 3b | L1 | 3 | 14,3223 | 0,65211 | 14,322693 | 0,652018 | 14,322699 | 0,6519960 | 14,3227000 | 0,6519987 |
| 14,404200000 | 14,322700000 | 0,733500000 | 0,652000000 | 1 | 0 | 0 | 3 | n | 50 | y | 3i | L1 | 3 | 14,4050 | 0,73380 | 14,404126 | 0,733513 | 14,404213 | 0,7334976 | 14,4042028 | 0,7334995 |
| 14,436800000 | 14,404200000 | 0,766100000 | 0,733500000 | 1 | 1 | 0 | 3 | n | 20 | y | 3j | L1 | 3 | 14,4377 | 0,76643 | 14,436780 | 0,766093 | 14,436785 | 0,7661042 | 14,4367965 | 0,7660987 |
| 14,518300000 | 14,436800000 | 0,847600000 | 0,766100000 | 0 | 1 | 0 | 3 | n | 50 | y | 3d | L1 | 3 | 14,5177 | 0,84729 | 14,518228 | 0,847615 | 14,518295 | 0,8476042 | 14,5183006 | 0,8475994 |
| 14,550900000 | 14,518300000 | 0,880200000 | 0,847600000 | 0 | 0 | 0 | 3 | n | 20 | n | 3e | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 14,632400000 | 14,550900000 | 0,961700000 | 0,880200000 | 1 | 0 | 0 | 2 | n | 50 | n | 2l | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 14,713900000 | 14,632400000 | 1,043200000 | 0,961700000 | 0 | 1 | 0 | 2 | n | 50 | n | 2g | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 14,746500000 | 14,713900000 | 1,075800000 | 1,043200000 | 0 | 0 | 0 | 2 | n | 20 | y | 2b | L1 | 3 | 14,7473 | 1,07608 | 14,746555 | 1,075803 | 14,746504 | 1,0757981 | 14,7465028 | 1,0757997 |
| 14,762800000 | 14,746500000 | 1,092100000 | 1,075800000 | 1 | 0 | 0 | 3 | n | 10 | n | 3k | GL | 4 | 14,7617 | 1,09204 | 14,762772 | 1,092089 | 14,762789 | 1,0920948 | 14,7628030 | 1,0920988 |
| 14,876900000 | 14,762800000 | 1,206200000 | 1,092100000 | 0 | 1 | 0 | 4 | n | 70 | n | 4k | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 14,893200000 | 14,876900000 | 1,222500000 | 1,206200000 | 1 | 0 | 0 | 2 | n | 10 | y | 2h | GL | 4 | 14,8923 | 1,22255 | 14,893205 | 1,222506 | 14,893199 | 1,2224996 | 14,8931977 | 1,2225009 |
| 15,007300000 | 14,893200000 | 1,336600000 | 1,222500000 | 0 | 1 | 0 | 4 | n | 70 | n | 4k | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 15,088800000 | 15,007300000 | 1,418100000 | 1,336600000 | 1 | 0 | 0 | 2 | n | 50 | y | 2i | L1 | 3 | 15,0882 | 1,41832 | 15,088764 | 1,418084 | 15,088793 | 1,4180961 | 15,0887996 | 1,4180988 |
| 15,121400000 | 15,088800000 | 1,450700000 | 1,418100000 | 1 | 1 | 0 | 3 | n | 20 | n | 3m | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |

| lat last | lat prev | lon last | lon prev | tl | tp | tr | i | p | tim | o | op | cal | f | decoded lat | decoded lon | decoded 4-bit lat | decoded 4-bit lon | decoded 6-bit lat | decoded 6-bit lon | decoded 8-bit lat | decoded 8-bit lon |
|-------------|-------------|-------------|-------------|----|----|----|---|---|-----|---|-----|-----|---|-------------|-------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| 15,15400000 | 15,12140000 | 1,48330000 | 1,45070000 | 1 | 1 | 0 | 2 | n | 20 | y | 2j | L1 | 3 | 15,1535 | 1,48358 | 15,154070 | 1,483292 | 15,154018 | 1,4832985 | 15,1539970 | 1,4832999 |
| 15,23550000 | 15,15400000 | 1,56480000 | 1,48330000 | 0 | 1 | 0 | 3 | n | 50 | n | 3g | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 15,26810000 | 15,23550000 | 1,59740000 | 1,56480000 | 0 | 0 | 0 | 2 | n | 20 | y | 2b | L1 | 3 | 15,2674 | 1,59764 | 15,268097 | 1,597408 | 15,268108 | 1,5973991 | 15,2681011 | 1,5973997 |
| 15,34960000 | 15,26810000 | 1,67890000 | 1,59740000 | 1 | 0 | 0 | 3 | n | 50 | n | 3l | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 15,36590000 | 15,34960000 | 1,69520000 | 1,67890000 | 0 | 1 | 0 | 2 | n | 10 | n | 2f | GL | 4 | 15,3651 | 1,69523 | 15,365952 | 1,695183 | 15,365906 | 1,6951980 | 15,3658966 | 1,6951987 |
| 15,48000000 | 15,36590000 | 1,80930000 | 1,69520000 | 1 | 0 | 0 | 4 | n | 70 | n | 4u | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 15,49630000 | 15,48000000 | 1,82560000 | 1,80930000 | 0 | 1 | 0 | 2 | n | 10 | y | 2c | GL | 4 | 15,4969 | 1,82578 | 15,496250 | 1,825596 | 15,496317 | 1,8256035 | 15,4963033 | 1,8256000 |
| 15,61040000 | 15,49630000 | 1,93970000 | 1,82560000 | 1 | 0 | 0 | 4 | n | 70 | y | 4p | L1 | 3 | 15,6107 | 1,93970 | 15,610316 | 1,939705 | 15,610391 | 1,9397047 | 15,6103982 | 1,9396994 |
| 15,62670000 | 15,61040000 | 1,95600000 | 1,93970000 | 0 | 1 | 0 | 3 | n | 10 | n | 3f | GL | 4 | 15,6264 | 1,95669 | 15,626722 | 1,956009 | 15,626689 | 1,9559985 | 15,6267005 | 1,9559987 |
| 15,74080000 | 15,62670000 | 2,07010000 | 1,95600000 | 1 | 0 | 0 | 4 | n | 70 | n | 4u | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 15,75710000 | 15,74080000 | 2,08640000 | 2,07010000 | 0 | 1 | 0 | 2 | y | 10 | n | 2a | GP | 4 | 15,7582 | 2,08624 | 15,757021 | 2,086422 | 15,757100 | 2,0864040 | 15,7570977 | 2,0864000 |
| 15,87120000 | 15,75710000 | 2,20050000 | 2,08640000 | 1 | 0 | 0 | 4 | n | 70 | y | 4p | L1 | 3 | 15,8719 | 2,20073 | 15,871182 | 2,200491 | 15,871211 | 2,2005036 | 15,8711974 | 2,2005011 |
| 15,88750000 | 15,87120000 | 2,21680000 | 2,20050000 | 0 | 1 | 0 | 3 | y | 10 | n | 3a | GP | 4 | 15,8877 | 2,21679 | 15,887493 | 2,216789 | 15,887510 | 2,2167990 | 15,8875044 | 2,2167987 |
| 15,90380000 | 15,88750000 | 2,23310000 | 2,21680000 | 1 | 0 | 0 | 4 | y | 10 | n | 4a | GP | 4 | 15,9046 | 2,23336 | 15,903835 | 2,233119 | 15,903783 | 2,2330994 | 15,9038010 | 2,2331003 |
| 15,92010000 | 15,90380000 | 2,24940000 | 2,23310000 | 0 | 1 | 0 | 4 | n | 10 | n | 4i | GL | 4 | 15,9194 | 2,24910 | 15,920112 | 2,249380 | 15,920084 | 2,2493951 | 15,9200965 | 2,2494010 |
| 19,09860000 | - | 5,42790000 | - | 1 | - | - | 1 | n | - | n | 1e | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 19,11490000 | 19,09860000 | 5,44420000 | 5,42790000 | 0 | 1 | 0 | 2 | n | 10 | n | 2f | GL | 4 | 19,1160 | 5,44438 | 19,114949 | 5,444196 | 19,114892 | 5,4442038 | 19,1148991 | 5,4442004 |
| 19,13120000 | 19,11490000 | 5,46050000 | 5,44420000 | 1 | 0 | 0 | 4 | n | 10 | n | 4q | GL | 4 | 19,1322 | 5,46021 | 19,131285 | 5,460491 | 19,131210 | 5,4604954 | 19,1312031 | 5,4604990 |
| 19,14750000 | 19,13120000 | 5,47680000 | 5,46050000 | 0 | 1 | 0 | 4 | n | 10 | n | 4i | GL | 4 | 19,1477 | 5,47670 | 19,147567 | 5,476788 | 19,147505 | 5,4768000 | 19,1475008 | 5,4768000 |
| 19,16380000 | 19,14750000 | 5,49310000 | 5,47680000 | 1 | 0 | 1 | 4 | n | 10 | n | 4zb | L2 | 4 | 19,1648 | 5,49281 | 19,163759 | 5,493107 | 19,163782 | 5,4931006 | 19,1637968 | 5,4930992 |
| 19,18010000 | 19,16380000 | 5,50940000 | 5,49310000 | 0 | 1 | 0 | 4 | n | 10 | n | 4i | GL | 4 | 19,1795 | 5,50948 | 19,180185 | 5,509385 | 19,180117 | 5,5094053 | 19,1801025 | 5,5093992 |
| 23,48330000 | - | 9,81260000 | - | 1 | - | - | 1 | n | - | y | 1d | L1 | 3 | 23,4826 | 9,81276 | 23,483292 | 9,812616 | 23,483304 | 9,8125971 | 23,4832965 | 9,8126010 |
| 23,49960000 | 23,48330000 | 9,82890000 | 9,81260000 | 0 | 1 | 0 | 3 | n | 10 | y | 3c | GL | 4 | 23,4994 | 9,82904 | 23,499564 | 9,828893 | 23,499586 | 9,8288960 | 23,4996010 | 9,8288993 |
| 23,51590000 | 23,49960000 | 9,84520000 | 9,82890000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 23,5152 | 9,84504 | 23,515945 | 9,845185 | 23,515916 | 9,8452037 | 23,5159001 | 9,8451997 |
| 23,53220000 | 23,51590000 | 9,86150000 | 9,84520000 | 0 | 1 | 1 | 4 | n | 10 | y | 4v | L2 | 4 | 23,5311 | 9,86152 | 23,532182 | 9,861518 | 23,532199 | 9,8614959 | 23,5322027 | 9,8614991 |
| 23,54850000 | 23,53220000 | 9,87780000 | 9,86150000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 23,5479 | 9,87767 | 23,548419 | 9,877818 | 23,548488 | 9,8778035 | 23,5485037 | 9,8778002 |
| 27,16710000 | - | 13,49640000 | - | 0 | - | - | 1 | n | - | y | 1b | L1 | 3 | 27,1673 | 13,49615 | 27,167103 | 13,496400 | 27,167120 | 13,4963982 | 27,1671041 | 13,4964006 |
| 27,18340000 | 27,16710000 | 13,51270000 | 13,49640000 | 1 | 0 | 0 | 3 | n | 10 | y | 3h | GL | 4 | 27,1824 | 13,51259 | 27,183315 | 13,512692 | 27,183390 | 13,5127051 | 27,1833966 | 13,5126996 |
| 27,19970000 | 27,18340000 | 13,52900000 | 13,51270000 | 0 | 1 | 0 | 4 | n | 10 | n | 4i | GL | 4 | 27,1990 | 13,52912 | 27,199721 | 13,529017 | 27,199693 | 13,5290048 | 27,1996962 | 13,5289993 |
| 27,21600000 | 27,19970000 | 13,54530000 | 13,52900000 | 1 | 0 | 0 | 4 | n | 10 | n | 4q | GL | 4 | 27,2151 | 13,54520 | 27,215968 | 13,545299 | 27,216002 | 13,5453002 | 27,2160002 | 13,5453006 |
| 27,23230000 | 27,21600000 | 13,56160000 | 13,54530000 | 0 | 1 | 1 | 4 | n | 10 | n | 4x | L2 | 4 | 27,2332 | 13,56150 | 27,232339 | 13,561602 | 27,232305 | 13,5616035 | 27,2322979 | 13,5616011 |
| 27,24860000 | 27,23230000 | 13,57790000 | 13,56160000 | 1 | 0 | 0 | 4 | n | 10 | n | 4q | GL | 4 | 27,2477 | 13,57773 | 27,248621 | 13,577890 | 27,248615 | 13,5778984 | 27,2486038 | 13,5779003 |
| 30,36190000 | - | 16,69120000 | - | 0 | - | - | 1 | n | - | y | 1b | L1 | 3 | 30,3614 | 16,69102 | 30,361940 | 16,691175 | 30,361889 | 16,6911946 | 30,3618971 | 16,6911990 |
| 30,37820000 | 30,36190000 | 16,70750000 | 16,69120000 | 1 | 0 | 0 | 3 | n | 10 | y | 3h | GL | 4 | 30,3774 | 16,70756 | 30,378112 | 16,707510 | 30,378204 | 16,7075036 | 30,3782049 | 16,7074993 |
| 30,39450000 | 30,37820000 | 16,72380000 | 16,70750000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 30,3956 | 16,72363 | 30,394558 | 16,723782 | 30,394502 | 16,7238012 | 30,3944988 | 16,7238000 |
| 30,50860000 | 30,39450000 | 16,83790000 | 16,72380000 | 1 | 0 | 0 | 4 | n | 70 | n | 4u | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 30,52490000 | 30,50860000 | 16,85420000 | 16,83790000 | 0 | 1 | 1 | 2 | n | 10 | n | 2o | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 30,54120000 | 30,52490000 | 16,87050000 | 16,85420000 | 1 | 0 | 0 | 2 | n | 10 | n | 2k | GL | 4 | 30,5407 | 16,87073 | 30,541198 | 16,870509 | 30,541186 | 16,8705060 | 30,5412032 | 16,8704992 |
| 30,55750000 | 30,54120000 | 16,88680000 | 16,87050000 | 0 | 1 | 0 | 4 | n | 10 | n | 4i | GL | 4 | 30,5568 | 16,88702 | 30,557474 | 16,886808 | 30,557486 | 16,8868044 | 30,5574976 | 16,8868008 |
| 33,36110000 | - | 19,69040000 | - | 0 | - | - | 1 | y | - | y | 1a | GP | 4 | 33,3602 | 19,69017 | 33,361068 | 19,690376 | 33,361101 | 19,6904029 | 33,3610991 | 19,6904002 |
| 33,37740000 | 33,36110000 | 19,70670000 | 19,69040000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 33,3765 | 19,70689 | 33,377349 | 19,706725 | 33,377384 | 19,7067038 | 33,3774015 | 19,7066991 |
| 33,39370000 | 33,37740000 | 19,72300000 | 19,70670000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 33,3944 | 19,72313 | 33,393686 | 19,723023 | 33,393714 | 19,7229977 | 33,3937008 | 19,7230007 |
| 33,50780000 | 33,39370000 | 19,83710000 | 19,72300000 | 1 | 0 | 0 | 4 | n | 70 | y | 4p | L1 | 3 | 33,5071 | 19,83722 | 33,507782 | 19,837107 | 33,507794 | 19,8371056 | 33,5077961 | 19,8370993 |
| 33,52410000 | 33,50780000 | 19,85340000 | 19,83710000 | 0 | 1 | 1 | 3 | n | 10 | y | 3n | L1 | 3 | 33,5238 | 19,85313 | 33,524158 | 19,853398 | 33,524085 | 19,8533967 | 33,5240979 | 19,8533993 |

| lat last | lat prev | lon last | lon prev | tl | tp | tr | i | p | tim | o | op | cal | f | decoded lat | decoded lon | decoded 4-bit lat | decoded 4-bit lon | decoded 6-bit lat | decoded 6-bit lon | decoded 8-bit lat | decoded 8-bit lon |
|--------------|--------------|--------------|--------------|----|----|----|---|---|-----|---|----|-----|---|-------------|-------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| 33,540400000 | 33,524100000 | 19,869700000 | 19,853400000 | 1 | 0 | 0 | 3 | n | 10 | y | 3h | GL | 4 | 33,5397 | 19,86999 | 33,540435 | 19,869707 | 33,540406 | 19,8696961 | 33,5403998 | 19,8696998 |
| 33,556700000 | 33,540400000 | 19,886000000 | 19,869700000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 33,5556 | 19,88571 | 33,556777 | 19,885981 | 33,556698 | 19,8860033 | 33,5566996 | 19,8859993 |
| 36,132100000 | - | 22,461400000 | - | 1 | - | - | 1 | y | - | y | 1a | GP | 4 | 36,1319 | 22,46135 | 36,132048 | 22,461405 | 36,132111 | 22,4613999 | 36,1320956 | 22,4613987 |
| 36,148400000 | 36,132100000 | 22,477700000 | 22,461400000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 36,1490 | 22,47788 | 36,148439 | 22,477722 | 36,148411 | 22,4777006 | 36,1484045 | 22,4776990 |
| 36,164700000 | 36,148400000 | 22,494000000 | 22,477700000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 36,1645 | 22,49431 | 36,164701 | 22,493974 | 36,164683 | 22,4939939 | 36,1646992 | 22,4940013 |
| 36,278800000 | 36,164700000 | 22,608100000 | 22,494000000 | 0 | 1 | 0 | 4 | n | 70 | y | 4f | L1 | 3 | 36,2784 | 22,60821 | 36,278737 | 22,608104 | 36,278782 | 22,6081025 | 36,2788017 | 22,6080991 |
| 36,295100000 | 36,278800000 | 22,624400000 | 22,608100000 | 1 | 0 | 1 | 3 | n | 10 | n | 3q | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 36,311400000 | 36,295100000 | 22,640700000 | 22,624400000 | 0 | 1 | 0 | 2 | n | 10 | n | 2f | GL | 4 | 36,3126 | 22,64107 | 36,311355 | 22,640673 | 36,311395 | 22,6406981 | 36,3114034 | 22,6407006 |
| 36,327700000 | 36,311400000 | 22,657000000 | 22,640700000 | 1 | 0 | 0 | 4 | n | 10 | n | 4q | GL | 4 | 36,3278 | 22,65682 | 36,327786 | 22,656993 | 36,327705 | 22,6570028 | 36,3276975 | 22,6569985 |
| 38,723800000 | - | 25,053100000 | - | 1 | - | - | 1 | y | - | n | 1a | GP | 4 | 38,7240 | 25,05280 | 38,723839 | 25,053089 | 38,723816 | 25,0531005 | 38,7238013 | 25,0530998 |
| 38,740100000 | 38,723800000 | 25,069400000 | 25,053100000 | 0 | 1 | 0 | 4 | n | 10 | n | 4i | GL | 4 | 38,7399 | 25,06919 | 38,740101 | 25,069416 | 38,740084 | 25,0693945 | 38,7400998 | 25,0693989 |
| 38,756400000 | 38,740100000 | 25,085700000 | 25,069400000 | 1 | 0 | 0 | 4 | n | 10 | n | 4q | GL | 4 | 38,7567 | 25,08535 | 38,756313 | 25,085701 | 38,756388 | 25,0856940 | 38,7564049 | 25,0856988 |
| 38,870500000 | 38,756400000 | 25,199800000 | 25,085700000 | 0 | 1 | 0 | 4 | n | 70 | n | 4k | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 38,886800000 | 38,870500000 | 25,216100000 | 25,199800000 | 1 | 0 | 1 | 2 | n | 10 | y | 2p | L1 | 3 | 38,8873 | 25,21601 | 38,886746 | 25,216128 | 38,886798 | 25,2161026 | 38,8867996 | 25,2161004 |
| 38,903100000 | 38,886800000 | 25,232400000 | 25,216100000 | 0 | 1 | 0 | 3 | n | 10 | y | 3c | GL | 4 | 38,9035 | 25,23266 | 38,903018 | 25,232427 | 38,903108 | 25,2323973 | 38,9030987 | 25,2324003 |
| 38,919400000 | 38,903100000 | 25,248700000 | 25,232400000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 38,9199 | 25,24897 | 38,919399 | 25,248727 | 38,919411 | 25,2486956 | 38,9194032 | 25,2486987 |
| 41,185100000 | - | 27,514400000 | - | 0 | - | - | 1 | n | - | y | 1b | L1 | 3 | 41,1844 | 27,51470 | 41,185069 | 27,514410 | 41,185120 | 27,5143983 | 41,1851019 | 27,5143990 |
| 41,201400000 | 41,185100000 | 27,530700000 | 27,514400000 | 1 | 0 | 0 | 3 | n | 10 | y | 3h | GL | 4 | 41,2006 | 27,53088 | 41,201346 | 27,530702 | 41,201398 | 27,5307061 | 41,2013993 | 27,5307003 |
| 41,217700000 | 41,201400000 | 27,547000000 | 27,530700000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 41,2186 | 27,54725 | 41,217687 | 27,547022 | 41,217693 | 27,5470049 | 41,2177036 | 27,5470011 |
| 41,331800000 | 41,217700000 | 27,661100000 | 27,547000000 | 1 | 0 | 0 | 4 | n | 70 | y | 4p | L1 | 3 | 41,3312 | 27,66104 | 41,331778 | 27,661097 | 41,331807 | 27,6611052 | 41,3318039 | 27,6611002 |
| 41,348100000 | 41,331800000 | 27,677400000 | 27,661100000 | 0 | 1 | 1 | 3 | n | 10 | n | 3o | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 41,364400000 | 41,348100000 | 27,693700000 | 27,677400000 | 1 | 0 | 0 | 2 | n | 10 | n | 2k | GL | 4 | 41,3639 | 27,69339 | 41,364431 | 27,693703 | 41,364420 | 27,6937014 | 41,3643976 | 27,6937009 |
| 41,380700000 | 41,364400000 | 27,710000000 | 27,693700000 | 0 | 1 | 0 | 4 | n | 10 | n | 4i | GL | 4 | 41,3797 | 27,71006 | 41,380778 | 27,709996 | 41,380716 | 27,7100014 | 41,3807024 | 27,7099994 |
| 43,532300000 | - | 29,861600000 | - | 0 | - | - | 1 | y | - | y | 1a | GP | 4 | 43,5311 | 29,86182 | 43,532357 | 29,861576 | 43,532317 | 29,8615993 | 43,5322988 | 29,8616011 |
| 43,548600000 | 43,532300000 | 29,877900000 | 29,861600000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 43,5491 | 29,87760 | 43,548598 | 29,877919 | 43,548610 | 29,8779025 | 43,5486026 | 29,8778990 |
| 43,564900000 | 43,548600000 | 29,894200000 | 29,877900000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 43,5653 | 29,89393 | 43,564975 | 29,894175 | 43,564890 | 29,8942058 | 43,5649005 | 29,8941994 |
| 43,679000000 | 43,564900000 | 30,008300000 | 29,894200000 | 1 | 0 | 0 | 4 | n | 70 | n | 4u | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 43,695300000 | 43,679000000 | 30,024600000 | 30,008300000 | 0 | 1 | 1 | 2 | n | 10 | y | 2n | L1 | 3 | 43,6947 | 30,02439 | 43,695273 | 30,024579 | 43,695301 | 30,0246038 | 43,6952977 | 30,0245988 |
| 43,711600000 | 43,695300000 | 30,040900000 | 30,024600000 | 1 | 0 | 0 | 3 | n | 10 | y | 3h | GL | 4 | 43,7124 | 30,04120 | 43,711684 | 30,040874 | 43,711592 | 30,0409059 | 43,7116009 | 30,0408985 |
| 43,727900000 | 43,711600000 | 30,057200000 | 30,040900000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 43,7289 | 30,05691 | 43,727891 | 30,057227 | 43,727914 | 30,0571962 | 43,7278994 | 30,0571999 |
| 45,781700000 | - | 32,111000000 | - | 1 | - | - | 1 | y | - | y | 1a | GP | 4 | 45,7821 | 32,11133 | 45,781745 | 32,111008 | 45,781698 | 32,1109953 | 45,7816982 | 32,1109997 |
| 45,798000000 | 45,781700000 | 32,127300000 | 32,111000000 | 0 | 1 | 0 | 4 | n | 10 | y | 4d | GL | 4 | 45,7973 | 32,12686 | 45,798012 | 32,127303 | 45,797983 | 32,1273027 | 45,7979964 | 32,1272992 |
| 45,814300000 | 45,798000000 | 32,143600000 | 32,127300000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 45,8148 | 32,14338 | 45,814218 | 32,143577 | 45,814311 | 32,1436018 | 45,8143018 | 32,1436002 |
| 45,928400000 | 45,814300000 | 32,257700000 | 32,143600000 | 0 | 1 | 0 | 4 | n | 70 | n | 4k | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 45,944700000 | 45,928400000 | 32,274000000 | 32,257700000 | 1 | 0 | 1 | 2 | n | 10 | n | 2q | NO | 2 | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC | NO CALC |
| 45,961000000 | 45,944700000 | 32,290300000 | 32,274000000 | 0 | 1 | 0 | 2 | n | 10 | n | 2f | GL | 4 | 45,9609 | 32,28987 | 45,960928 | 32,290331 | 45,961007 | 32,2903019 | 45,9609953 | 32,2902990 |
| 45,977300000 | 45,961000000 | 32,306600000 | 32,290300000 | 1 | 0 | 0 | 4 | n | 10 | n | 4q | GL | 4 | 45,9780 | 32,30655 | 45,977304 | 32,306621 | 45,977293 | 32,3065994 | 45,9773001 | 32,3065983 |
| 47,949600000 | - | 34,278900000 | - | 1 | - | - | 1 | y | - | n | 1a | GP | 4 | 47,9498 | 34,27848 | 47,949585 | 34,278890 | 47,949602 | 34,2788967 | 47,9495964 | 34,2788981 |
| 47,965900000 | 47,949600000 | 34,295200000 | 34,278900000 | 0 | 1 | 0 | 4 | n | 10 | n | 4i | GL | 4 | 47,9658 | 34,29500 | 47,965986 | 34,295197 | 47,965891 | 34,2951929 | 47,9658985 | 34,2951992 |
| 47,982200000 | 47,965900000 | 34,311500000 | 34,295200000 | 1 | 0 | 0 | 4 | n | 10 | n | 4q | GL | 4 | 47,9824 | 34,31192 | 47,982238 | 34,311510 | 47,982215 | 34,3115032 | 47,9822000 | 34,3115018 |
| 48,096300000 | 47,982200000 | 34,425600000 | 34,311500000 | 0 | 1 | 0 | 4 | n | 70 | y | 4f | L1 | 3 | 48,0952 | 34,42593 | 48,096285 | 34,425602 | 48,096302 | 34,4256044 | 48,0962957 | 34,4256013 |
| 48,112600000 | 48,096300000 | 34,441900000 | 34,425600000 | 1 | 0 | 1 | 3 | n | 10 | y | 3p | L1 | 3 | 48,1130 | 34,44224 | 48,112671 | 34,441887 | 48,112584 | 34,4419050 | 48,1126046 | 34,4419012 |
| 48,128900000 | 48,112600000 | 34,458200000 | 34,441900000 | 0 | 1 | 0 | 3 | n | 10 | y | 3c | GL | 4 | 48,1294 | 34,45809 | 48,128903 | 34,458231 | 48,128914 | 34,4582020 | 48,1288973 | 34,4581995 |

| lat last | lat prev | lon last | lon prev | tl | tp | tr | i | p | tim | o | op | cal | f | decoded lat | decoded lon | decoded 4-bit lat | decoded 4-bit lon | decoded 6-bit lat | decoded 6-bit lon | decoded 8-bit lat | decoded 8-bit lon |
|--------------|--------------|--------------|--------------|----|----|----|---|---|-----|---|----|-----|---|-------------|-------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| 48,145200000 | 48,128900000 | 34,474500000 | 34,458200000 | 1 | 0 | 0 | 4 | n | 10 | y | 4l | GL | 4 | 48,1457 | 34,47421 | 48,145145 | 34,474491 | 48,145197 | 34,4744955 | 48,1451983 | 34,4745005 |

7.4.3.1.4 Tables of values for use in content checking test cases

The values in table 7.8 are used in the test "ADSB_Basic_A".

Table 7.8: Values for use in the test "ADSB_BASIC_A"

| ADSB_BASIC_APPIN_PARAMETERS(x) | | | | | ADSB_BASIC_RF_OUT_PARAMETERS(x) | | | |
|---|--|--|--|--|---------------------------------|-------------------------------|-------------------------------|-------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| sil | Probability of exceeding the R_c integrity containment radius without detection: unknown | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-3} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-5} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-7} per flight hour or per operation | 0 | 1 | 2 | 3 |
| lat6 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| br/gr | Barometric altitude rate | Barometric altitude rate | Barometric altitude rate | Geometric altitude rate | 0 | 0 | 0 | 1 |
| lon6 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| altr bit 9 | Climbing | Climbing | Descending | Descending | 0 | 0 | 1 | 1 |
| altr bits 1-8 | 20 000 fpm | 0 fpm | 32 000 fpm | Unknown | 201 | 1 | 255 | 0 |
| bgo | 4 000 ft | -2 000 ft | -5 000 ft | 7 200 ft | 123 | 3 | 2 | 124 |
| gs | 1 022 | 0 | 3 069 | unknown | 1 023 | 1 | 2 047 | 0 |
| gt | 51 degrees | 0 degrees (north) | 359 degrees | 0 degrees (north) | 290 | 0 | 2 042 | 0 |
| Assumed altitude to calculate baro/geo offset (bgo) from (ft) | 20 000 feet | 20 000 feet | 30 000 feet | 30 000 feet | N/A | N/A | N/A | N/A |

The values in table 7.9 are used in the test "ADSB_Basic_Rec_A".

Table 7.9: Values for use in the test "ADSB_Basic_Rec_A"

| ADSB_BASIC_RF_IN_PARAMETERS(x) | | | | | ADSB_BASIC_APOUT_PARAMETERS(x) | | | |
|---|-------------------------------|-------------------------------|-------------------------------|-------------------------------|--|--|--|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_B | add_B | add_B | add_B | add_B | add_B | add_B | add_B |
| sil | 0 | 1 | 2 | 3 | Probability of exceeding the R_c integrity containment radius without detection: unknown | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-3} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-5} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-7} per flight hour or per operation |
| lat6 | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode |
| br/gr | 0 | 0 | 0 | 1 | Barometric altitude rate | Barometric altitude rate | Barometric altitude rate | Geometric altitude rate |
| lon6 | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode |
| altr bit 9 | 0 | 0 | 1 | 1 | Climbing | Climbing | Descending | Descending |
| altr bits 1-8 | 201 | 1 | 255 | 0 | 20 000 fpm | 0 fpm | 32 000 fpm | Unknown |
| bgo | 123 | 3 | 2 | 124 | 4 000 ft | -2 000 ft | -5 000 ft | 7 200 ft |
| gs | 1 023 | 1 | 2 047 | 0 | 1 022 | 0 | 3 069 | unknown |
| gt | 290 | 0 | 2 042 | 0 | 51 degrees | 0 degrees (north) | 359 degrees | 0 degrees (north) |
| Assumed altitude to calculate baro/geo offset (bgo) from (ft) | N/A | N/A | N/A | N/A | 20 000 feet | 20 000 feet | 30 000 feet | 30 000 feet |

The values in table 7.10 are used in the test "ADSB_High_Dynamic_A".

Table 7.10: Values for use in the test "ADSB_High_Dynamic_A"

| ADSB_HIGH_DYNAMIC_APPIN_PARAMETERS(x) | | | | | ADSB_HIGH_DYNAMIC_RF_OUT_PARAMETERS(x) | | | |
|---|--|--|--|--|--|-------------------------------|-------------------------------|-------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| lat4 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| br/gr | Barometric altitude rate | Barometric altitude rate | Barometric altitude rate | Geometric altitude rate | 0 | 0 | 0 | 1 |
| lon4 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| altr bit 9 | Climbing | Climbing | Descending | Descending | 0 | 0 | 1 | 1 |
| altr bits 1-8 | 20 000 fpm | 0 fpm | 32 000 fpm | Unknown | 201 | 1 | 255 | 0 |
| sil | Probability of exceeding the R_c integrity containment radius without detection: unknown | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-3} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-5} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-7} per flight hour or per operation | 0 | 1 | 2 | 3 |
| bgo | 4 000 ft | -2 000 ft | -5 000 ft | 7 200 ft | 123 | 3 | 2 | 124 |
| gs | 1 022 | 0 | 3 069 | unknown | 1 023 | 1 | 2 047 | 0 |
| gt | 51 degrees | 0 degrees (north) | 359 degrees | 0 degrees (north) | 290 | 0 | 2 042 | 0 |
| Assumed altitude to calculate baro/geo offset (bgo) from (ft) | 20 000 feet | 20 000 feet | 30 000 feet | 30 000 feet | N/A | N/A | N/A | N/A |

The values in table 7.11 are used in the test "ADSB_High_Dynamic_Rec_A".

Table 7.11: Values for use in the test "ADSB_High_Dynamic_Rec_A"

| ADSB_HIGH_DYNAMIC_RF_IN_PARAMETERS(x) | | | | | ADSB_HIGH_DYNAMIC_APOUT_PARAMETERS(x) | | | |
|---|---|---|---|---|--|--|--|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_B | add_B | add_B | add_B | add_B | add_B | add_B | add_B |
| lat4 | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| br/gr | 0 | 0 | 0 | 1 | Barometric altitude rate | Barometric altitude rate | Barometric altitude rate | Geometric altitude rate |
| lon4 | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| altr bit 9 | 0 | 0 | 1 | 1 | Climbing | Climbing | Descending | Descending |
| altr bits 1-8 | 201 | 1 | 255 | 0 | 20 000 fpm | 0 fpm | 32 000 fpm | Unknown |
| sil | 0 | 1 | 2 | 3 | Probability of exceeding the R_c integrity containment radius without detection: unknown | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-3} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-5} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-7} per flight hour or per operation |
| bgo | 123 | 3 | 2 | 124 | 4 000 ft | -2 000 ft | -5 000 ft | 7 200 ft |
| gs | 1 023 | 1 | 2 047 | 0 | 1 022 | 0 | 3 069 | unknown |
| gt | 290 | 0 | 2 042 | 0 | 51 degrees | 0 degrees (north) | 359 degrees | 0 degrees (north) |
| Assumed altitude to calculate baro/geo offset (bgo) from (ft) | N/A | N/A | N/A | N/A | 20 000 feet | 20 000 feet | 30 000 feet | 30 000 feet |

The values in table 7.12 are used in the test "ADSB_Full_Position_A".

Table 7.12: Values for use in the test "ADSB_Full_Position_A"

| ADSB_FULL_POSITION_APPIN_PARAMETERS(x) | | | | | ADSB_FULL_POSITION_RF_OUT_PARAMETERS(x) | | | |
|---|--|--|--|--|---|---|---|---|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| lat6 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encod e | As encoded in test CPR_Encod e | As encoded in test CPR_Encod e | As encoded in test CPR_Encod e |
| pid | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encod e | As encoded in test CPR_Encod e | As encoded in test CPR_Encod e | As encoded in test CPR_Encod e |
| lon6 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encod e | As encoded in test CPR_Encod e | As encoded in test CPR_Encod e | As encoded in test CPR_Encod e |
| bgo | 4 000 ft | -2 000 ft | -5 000 ft | 7 200 ft | 123 | 3 | 2 | 124 |
| sil | Probability of exceeding the R_c integrity containment radius without detection: unknown | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-3} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-5} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-7} per flight hour or per operation | 0 | 1 | 2 | 3 |
| gs | 1 022 | 0 | 3 069 | unknown | 1 023 | 1 | 2 047 | 0 |
| gt | 51 degrees | 0 degrees (north) | 359 degrees | 0 degrees (north) | 290 | 0 | 2 042 | 0 |
| Assumed altitude to calculate baro/geo offset (bgo) from (ft) | 20 000 feet | 20 000 feet | 30 000 feet | 30 000 feet | N/A | N/A | N/A | N/A |

The values in table 7.13 are used in the test "ADSB_Full_Position_Rec_A".

Table 7.13: Values for use in the test "ADSB_Full_Position_Rec_A"

| ADSB_FULL_POSITION_RF_IN_PARAMETERS(x) | | | | | ADSB_FULL_POSITION_APPOUT_PARAMETERS(x) | | | |
|---|---|---|---|---|--|--|--|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_B | add_B | add_B | add_B | add_B | add_B | add_B | add_B |
| lat6 | As decoded in test CPR_Deco de | As decoded in test CPR_Deco de | As decoded in test CPR_Deco de | As decoded in test CPR_Deco de | As decoded in test CPR_Deco | As decoded in test CPR_Deco | As decoded in test CPR_Deco | As decoded in test CPR_Deco |
| pid | As decoded in test CPR_Deco de | As decoded in test CPR_Deco de | As decoded in test CPR_Deco de | As decoded in test CPR_Deco de | As decoded in test CPR_Deco | As decoded in test CPR_Deco | As decoded in test CPR_Deco | As decoded in test CPR_Deco |
| lon6 | As decoded in test CPR_Deco de | As decoded in test CPR_Deco de | As decoded in test CPR_Deco de | As decoded in test CPR_Deco de | As decoded in test CPR_Deco | As decoded in test CPR_Deco | As decoded in test CPR_Deco | As decoded in test CPR_Deco |
| bgo | 123 | 3 | 2 | 124 | 4 000 ft | -2 000 ft | -5 000 ft | 7 200 ft |
| sil | 0 | 1 | 2 | 3 | Probability of exceeding the R_c integrity containment radius without detection: unknown | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-3} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-5} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-7} per flight hour or per operation |
| gs | 1 023 | 1 | 2 047 | 0 | 1 022 | 0 | 3 069 | unknown |
| gt | 290 | 0 | 2 042 | 0 | 51 degrees | 0 degrees (north) | 359 degrees | 0 degrees (north) |
| Assumed altitude to calculate baro/geo offset (bgo) from (ft) | N/A | N/A | N/A | N/A | 20 000 feet | 20 000 feet | 30 000 feet | 30 000 feet |

The values in table 7.14 are used in the test "ADSB_Basic_Ground_Rec_A".

Table 7.14: Values for use in the test "ADSB_Basic_Ground_Rec_A"

| ADSB_BASIC_GROUND_RF_IN_PARAMETERS(x) | | | | | ADSB_BASIC_GROUND_APOUT_PARAMETERS(x) | | | |
|---|---|---|---|---|---------------------------------------|----------------------------------|----------------------------------|----------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| h | 5 | 1 | 23 | 0 | 5am | 1am | 11pm | midnight |
| min | 27 | 1 | 59 | 0 | 27 minutes | 1 minute | 59 minutes | 0 minutes |
| pid | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| bgo | 123 | 3 | 2 | 124 | 4 000 ft | -2 000 ft | -5 000 ft | 7 200 ft |
| slt | 49 | 1 | 255 | 0 | slot 50 | slot 2 | slot 256 | first slot in second frame |
| lat4 | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| lon4 | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Enco de | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| Assumed altitude to calculate baro/geo offset (bgo) from (ft) | N/A | N/A | N/A | N/A | 20 000 feet | 20 000 feet | 30 000 feet | 30 000 feet |

The values in table 7.15 are used in the test "ADSB.UTC_Time_Rec_A".

Table 7.15: Values for use in the test "ADSB.UTC_Time_Rec_A"

| ADSB.UTC_TIME_RF_IN_PARAMETERS(x) | | | | | ADSB.UTC_TIME_APPOUT_PARAMETERS(x) | | | |
|-----------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|------------------------------------|-------------------------------|-------------------------------|-------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| day | 6 | 1 | 31 | 1 | day 6 | day 1 | day 31 | day 1 |
| year | 33 | 34 | 33 | 35 | 2003 | 2004 | 2003 | 2005 |
| h | 5 | 1 | 23 | 0 | 5 am | 1 am | 11 pm | midnight |
| min | 27 | 1 | 59 | 0 | 27 minutes | 1 minute | 59 minutes | 0 minutes |
| slt | 49 | 1 | 255 | 0 | slot 50 | slot 2 | slot 256 | first slot in second frame |
| lat4 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| lon4 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| sec | 25 | 0 | 60 | 0 | 25 seconds | 0 seconds | 60 seconds | 0 seconds |

The values in table 7.16 are used in the test "ADSB_Two_Slot_TCP/SVQ_A".

Table 7.16: Values for use in the test "ADSB_Two_Slot_TCP/SVQ_A"

| ADSB.TWO_SLOT_TCP/SVQ_APPIN_PARAMETERS(x) | | | | | ADSB.TWO_SLOT_TCP/SVQ_RF_OUT_PARAMETERS(x) | | | |
|---|-------------------------------|-------------------------------|-------------------------------|-------------------------------|--|-------------------------------|-------------------------------|-------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| lat | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| balt | 8 025 feet | -1 300 feet | 130 000 feet | 0 feet | 936 | 2 | 4 072 | 132 |
| lon | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| ttg | 0,5 minutes | 0 minutes | 15,25 minutes | 0 minutes | 3 | 0 | 62 | 0 |
| lat | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| balt | 8 025 feet | -1 300 feet | 130 000 feet | 0 feet | 936 | 2 | 4 072 | 132 |

| ADSB_TWO_SLOT_TCP/SVQ_APPIN_PARAMETERS(x) | | | | | ADSB_TWO_SLOT_TCP/SVQ_RF_OUT_PARAMETERS(x) | | | |
|---|---|---|---|---|--|-------------------------------|-------------------------------|-------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| lon | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| ttg | 0,75 minutes | 0 minutes | 15,25 minutes | 0 minutes | 4 | 0 | 62 | 0 |
| lat | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| balt | 8 025 feet | -1 300 feet | 130 000 feet | 0 feet | 936 | 2 | 4 072 | 132 |
| lon | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| ttg | 1,0 minutes | 0 minutes | 15,25 minutes | 0 minutes | 5 | 0 | 62 | 0 |
| lat | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| balt | 8 025 feet | -1 300 feet | 130 000 feet | 0 feet | 936 | 2 | 4 072 | 132 |
| lon | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| ttg | 1,25 minutes | 0 minutes | 15,25 minutes | 0 minutes | 6 | 0 | 62 | 0 |
| csi | AAAA | BBBB | CCCC | DDDD | 00 00 00 00 | 01 01 01 01 | 02 02 02 02 | 03 03 03 03 |
| st | no emergency/not reported | general emergency | no communications | unlawful interference | 0 | 1 | 4 | 5 |
| csr | 0000 | 1111 | 2222 | 3333 | 26 26 26 26 | 27 27 27 27 | 28 28 28 28 | 29 29 29 29 |
| nacv | Horizontal velocity error (95 %) < 10 m/s | Horizontal velocity error (95 %) < 3 m/s | Horizontal velocity error (95 %) < 1 m/s | Horizontal velocity error (95 %) < 0,3 m/s | 1 | 2 | 3 | 4 |
| nacp | EPU < 7,408 Km (4 NM) | EPU ≥ 18,52 Km (10 NM) | EPU < 555,6 m (0,3 NM) | EPU < 3 m and VEPU < 4 m | 2 | 0 | 6 | 11 |
| cdti | CDTI equipment installed and operational | CDTI equipment installed and operational | not CDTI equipped | not CDTI equipped | 1 | 1 | 0 | 0 |
| nicb | Barometric pressure altitude has been cross checked against another source of pressure altitude | Barometric pressure altitude has not been cross checked against another source of pressure altitude | Barometric pressure altitude has been cross checked against another source of pressure altitude | Barometric pressure altitude has not been cross checked against another source of pressure altitude | 1 | 0 | 1 | 0 |
| acas | ACAS installed and operational | ACAS installed and operational | ACAS installed and operational | ACAS installed and operational | 1 | 1 | 1 | 1 |
| ra | resolution activity active | resolution activity not active | resolution activity active | resolution activity not active | 1 | 1 | 1 | 0 |

| ADSB_TWO_SLOT_TCP/SVQ_APPIN_PARAMETERS(x) | | | | | ADSB_TWO_SLOT_TCP/SVQ_RF_OUT_PARAMETERS(x) | | | |
|---|---|---|---|---|--|-------------------------------|-------------------------------|-------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| pid | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| lat6 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| nacv | Horizontal velocity error unknown | Horizontal velocity error < 10 m/s | Horizontal velocity error < 3 m/s | Horizontal velocity error < 1 m/s | 0 | 1 | 2 | 3 |
| lon6 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| br/gr | Barometric altitude rate | Barometric altitude rate | Barometric altitude rate | Geometric altitude rate | 0 | 0 | 0 | 1 |
| altr bit 9 | Climbing | Climbing | Descending | Descending | 0 | 0 | 1 | 1 |
| altr bits 1 to 8 | 20 000 fpm | 0 fpm | 32 000 fpm | Unknown | 201 | 1 | 255 | 0 |
| ac | heavy a/c | medium a/c | light a/c | medium a/c | 4 | 2 | 0 | 2 |
| sil | Probability of exceeding the R _c integrity containment radius without detection: unknown | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻³ per flight hour or per operation | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻⁵ per flight hour or per operation | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻⁷ per flight hour or per operation | 0 | 1 | 2 | 3 |
| gs | 1 022 | 0 | 3 069 | unknown | 1 023 | 1 | 2 047 | 0 |
| gt | 51 degrees | 0 degrees (north) | 359 degrees | 0 degrees (north) | 290 | 0 | 2 042 | 0 |

The values in table 7.17 are used in the test "ADSB_Two_Slot_TCP/SVQ_Rec_A".

Table 7.17: Values for use in the test "ADSB_Two_Slot_TCP/SVQ_Rec_A"

| ADSB_TWO_SLOT_TCP/SVQ_RF_IN_PARAMETERS(x) | | | | | ADSB_TWO_SLOT_TCP/SVQ_APOUT_PARAMETERS(x) | | | |
|---|-------------------------------|-------------------------------|-------------------------------|-------------------------------|---|-------------------------------|-------------------------------|-------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_B | add_B | add_B | add_B | add_B | add_B | add_B | add_B |
| lat | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| balt | 936 | 2 | 4 072 | 132 | 8 025 feet | -1 300 feet | 130 000 feet | 0 feet |
| lon | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| ttg | 3 | 0 | 62 | 0 | 0,5 minutes | 0 minutes | 15,25 minutes | 0 minutes |
| lat | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| balt | 936 | 2 | 4 072 | 132 | 8 025 feet | -1 300 feet | 130 000 feet | 0 feet |
| lon | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| ttg | 4 | 0 | 62 | 0 | 0,75 minutes | 0 minutes | 15,25 minutes | 0 minutes |
| lat | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| balt | 936 | 2 | 4 072 | 132 | 8 025 feet | -1 300 feet | 130 000 feet | 0 feet |
| lon | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| ttg | 5 | 0 | 62 | 0 | 1,0 minutes | 0 minutes | 15,25 minutes | 0 minutes |
| lat | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| balt | 936 | 2 | 4 072 | 132 | 8 025 feet | -1 300 feet | 130 000 feet | 0 feet |
| lon | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| ttg | 6 | 0 | 62 | 0 | 1,25 minutes | 0 minutes | 15,25 minutes | 0 minutes |
| csl | 00 00 00 00 | 01 01 01 01 | 02 02 02 02 | 03 03 03 03 | AAAA | BBBB | CCCC | DDDD |
| st | 0 | 1 | 4 | 5 | no emergency/not reported | general emergency | no communications | unlawful interference |
| csr | 26 26 26 26 | 27 27 27 27 | 28 28 28 28 | 29 29 29 29 | 0000 | 1111 | 2222 | 3333 |

| ADSB_TWO_SLOT_TCP/SVQ_RF_IN_PARAMETERS(x) | | | | | ADSB_TWO_SLOT_TCP/SVQ_APOUT_PARAMETERS(x) | | | |
|---|-------------------------------|-------------------------------|-------------------------------|-------------------------------|---|---|---|---|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| nacv | 1 | 2 | 3 | 4 | Horizontal velocity error (95 %) < 10 m/s | Horizontal velocity error (95 %) < 3 m/s | Horizontal velocity error (95 %) < 1 m/s | Horizontal velocity error (95 %) < 0,3 m/s |
| nacp | 2 | 0 | 6 | 11 | EPU < 7,408 Km (4 NM) | EPU ≥ 18,52 Km (10 NM) | EPU < 555,6 m (0,3 NM) | EPU < 3 m and VEPU < 4 m |
| cdti | 1 | 1 | 0 | 0 | CDTI equipment installed and operational | CDTI equipment installed and operational | not CDTI equipped | not CDTI equipped |
| nicb | 1 | 0 | 1 | 0 | Barometric pressure altitude has been cross checked against another source of pressure altitude | Barometric pressure altitude has not been cross checked against another source of pressure altitude | Barometric pressure altitude has been cross checked against another source of pressure altitude | Barometric pressure altitude has not been cross checked against another source of pressure altitude |
| acas | 1 | 1 | 1 | 1 | ACAS installed and operational | ACAS installed and operational | ACAS installed and operational | ACAS installed and operational |
| ra | 1 | 1 | 1 | 0 | resolution activity active | resolution activity not active | resolution activity active | resolution activity not active |
| pid | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| lat6 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| nacv | 0 | 1 | 2 | 3 | Horizontal velocity error unknown | Horizontal velocity error < 10 m/s | Horizontal velocity error < 3 m/s | Horizontal velocity error < 1 m/s |
| lon6 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| br/gr | 0 | 0 | 0 | 1 | Barometric altitude rate | Barometric altitude rate | Barometric altitude rate | Geometric altitude rate |
| altr bit 9 | 0 | 0 | 1 | 1 | Climbing | Climbing | Descending | Descending |
| altr bits 1-8 | 201 | 1 | 255 | 0 | 20 000 fpm | 0 fpm | 32 000 fpm | Unknown |
| ac | 4 | 2 | 0 | 2 | heavy a/c | medium a/c | light a/c | medium a/c |
| sil | 0 | 1 | 2 | 3 | Probability of exceeding the R _c integrity containment radius without detection: unknown | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻³ per flight hour or per operation | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻⁵ per flight hour or per operation | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻⁷ per flight hour or per operation |

| ADSB_TWO_SLOT_TCP/SVQ_RF_IN_PARAMETERS(x) | | | | | ADSB_TWO_SLOT_TCP/SVQ_APPOUT_PARAMETERS(x) | | | |
|---|-------------------------|----------------|----------------|-----------------|--|-------------------|----------------|-------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| gs | 1 023 | 1 | 2 047 | 0 | 1 022 | 0 | 3 069 | unknown |
| gt | 290 | 0 | 2 042 | 0 | 51 degrees | 0 degrees (north) | 359 degrees | 0 degrees (north) |

The values in table 7.18 are used in the test "ADSB_Single_Slot_TCP_A".

Table 7.18: Values for use in the test "ADSB_Single_Slot_TCP_A"

| ADSB_SINGLE_SLOT_TCP_APPIN_PARAMETERS(x) | | | | | ADSB_SINGLE_SLOT_TCP_RF_OUT_PARAMETERS(x) | | | |
|--|----------------------------------|----------------------------------|----------------------------------|----------------------------------|---|----------------------------------|----------------------------------|----------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| balt | 8 025 feet | -1 300 feet | 130 000 feet | 0 feet | 936 | 2 | 4 072 | 132 |
| no | Current TCP | Next TCP | Next +1 TCP | Next +2 TCP | 0 | 1 | 2 | 3 |
| Patch ID | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| lat | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| lon | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| ttg | 0,75 minutes | 0 minutes | 15,25 minutes | 0 minutes | 4 | 0 | 62 | 0 |

The values in table 7.19 are used in the test "ADSB_Single_Slot_TCP_Rec_A".

Table 7.19: Values for use in the test "ADSB_Single_Slot_TCP_Rec_A"

| ADSB_SINGLE_SLOT_TCP_RF_IN_PARAMETERS(x) | | | | | ADSB_SINGLE_SLOT_TCP_APOUT_PARAMETERS(x) | | | |
|--|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|--|----------------------------------|----------------------------------|----------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_B | add_B | add_B | add_B | add_B | add_B | add_B | add_B |
| balt | 936 | 2 | 4 072 | 132 | 8 025 feet | -1 300 feet | 130 000 feet | 0 feet |
| no | 0 | 1 | 2 | 3 | Current TCP | Next TCP | Next +1 TCP | Next +2 TCP |
| Patch ID | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| lat | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| lon | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| ttg | 4 | 0 | 62 | 0 | 0,75 minutes | 0 minutes | 15,25 minutes | 0 minutes |

The values in table 7.20 are used in the test "ADSB_Single_Slot_SVQ_A".

Table 7.20: Values for use in the test "ADSB_Single_Slot_SVQ_A"

| ADSB_SINGLE_SLOT_SVQ_APPIN_PARAMETERS(x) | | | | | ADSB_SINGLE_SLOT_SVQ_RF_OUT_PARAMETERS(x) | | | |
|--|---|---|---|---|---|----------------|----------------|-----------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| NACp | EPU < 7,408 Km (4 NM) | EPU ≥ 18,52 Km (10 NM) | EPU < 555,6 m (0,3 NM) | EPU < 3 m and VEPU < 4 m | 2 | 0 | 6 | 11 |
| Sil | Probability of exceeding the R _c integrity containment radius without detection: unknown | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻³ per flight hour or per operation | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻⁵ per flight hour or per operation | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻⁷ per flight hour or per operation | 0 | 1 | 2 | 3 |
| nicb | Barometric pressure altitude has been cross checked against another source of pressure altitude | Barometric pressure altitude has not been cross checked against another source of pressure altitude | Barometric pressure altitude has been cross checked against another source of pressure altitude | Barometric pressure altitude has not been cross checked against another source of pressure altitude | 1 | 0 | 1 | 0 |
| acas | ACAS installed and operational | ACAS installed and operational | ACAS installed and operational | ACAS installed and operational | 1 | 1 | 1 | 1 |
| ra | resolution activity active | resolution activity not active | resolution activity active | resolution activity not active | 1 | 1 | 1 | 0 |
| st | no emergency/not reported | general emergency | no communications | unlawful interference | 0 | 1 | 4 | 5 |
| ac | rotorcraft | unmanned aerial vehicle | medium a/c | medium a/c | 9 | 12 | 2 | 2 |

The values in table 7.21 are used in the test "ADSB_Single_Slot_SVQ_Rec_A".

Table 7.21: Values for use in the test "ADSB_Single_Slot_SVQ_Rec_A"

| ADSB_SINGLE_SLOT_SVQ_RF_IN_PARAMETERS(x) | | | | | ADSB_SINGLE_SLOT_SVQ_APPOUT_PARAMETERS(x) | | | |
|--|----------------------------|----------------|----------------|-----------------|---|---|---|---|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_B | add_B | add_B | add_B | add_B | add_B | add_B | add_B |
| nacp | 2 | 0 | 6 | 11 | EPU < 7,408 Km (4 NM) | EPU ≥ 18,52 Km (10 NM) | EPU < 555,6m (0,3 NM) | EPU < 3 m and VEPU < 4 m |
| sil | 0 | 1 | 2 | 3 | Probability of exceeding the R _c integrity containment radius without detection: unknown | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻³ per flight hour or per operation | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻⁵ per flight hour or per operation | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻⁷ per flight hour or per operation |
| nicb | 1 | 0 | 1 | 0 | Barometric pressure altitude has been cross checked against another source of pressure altitude | Barometric pressure altitude has not been cross checked against another source of pressure altitude | Barometric pressure altitude has been cross checked against another source of pressure altitude | Barometric pressure altitude has not been cross checked against another source of pressure altitude |
| acas | 1 | 1 | 1 | 1 | ACAS installed and operational | ACAS installed and operational | ACAS installed and operational | ACAS installed and operational |
| ra | 1 | 1 | 1 | 0 | resolution activity active | resolution activity not active | resolution activity active | resolution activity not active |
| st | 0 | 1 | 4 | 5 | no emergency/not reported | general emergency | no communications | unlawful interference |
| ac | 9 | 12 | 2 | 2 | rotorcraft | unmanned aerial vehicle | medium a/c | medium a/c |

The values in table 7.22 are used in the test "ADSB_Aircraft_Data_A".

Table 7.22: Values for use in the test "ADSB_Single_Slot_SVQ_Rec_A"

| ADSB_AIRCRAFT_DATA_APPIN_PARAMETERS(x) | | | | | ADSB_AIRCRAFT_DATA_RF_OUT_PARAMETERS(x) | | | |
|--|---------------------------|-------------------------|-------------------|-----------------------|---|----------------|----------------|-----------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| ac | rotorcraft | unmanned aerial vehicle | medium a/c | medium a/c | 9 | 12 | 2 | 2 |
| st | no emergency/not reported | general emergency | no communications | unlawful interference | 0 | 1 | 4 | 5 |
| csl | AAAA | BBBB | CCCC | DDDD | 00 00 00 00 | 01 01 01 01 | 02 02 02 02 | 03 03 03 03 |
| csr | 0000 | 1111 | 2222 | 3333 | 26 26 26 26 | 27 27 27 27 | 28 28 28 28 | 29 29 29 29 |

The values in table 7.23 are used in the test "ADSB_Aircraft_Data_Rec_A".

Table 7.23: Values for use in the test "ADSB_Aircraft_Data_Rec_A"

| ADSB_AIRCRAFT_DATA_RF_IN_PARAMETERS(x) | | | | | ADSB_AIRCRAFT_DATA_APPOUT_PARAMETERS(x) | | | |
|--|-------------------------|----------------|----------------|-----------------|---|-------------------------|-------------------|-----------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_B | add_B | add_B | add_B | add_B | add_B | add_B | add_B |
| ac | 9 | 12 | 2 | 2 | rotorcraft | unmanned aerial vehicle | medium a/c | medium a/c |
| st | 0 | 1 | 4 | 5 | no emergency/not reported | general emergency | no communications | unlawful interference |
| csl | 00 00 00 00 | 01 01 01 01 | 02 02 02 02 | 03 03 03 03 | AAAA | BBBB | CCCC | DDDD |
| csr | 26 26 26 26 | 27 27 27 27 | 28 28 28 28 | 29 29 29 29 | 0000 | 1111 | 2222 | 3333 |

The values in table 7.24 are used in the test "ADSB_High_Resolution_A".

Table 7.24: Values for use in the test "ADSB_High_Resolution_A"

| ADSB_HIGH_RESOLUTION_APPIN_PARAMETERS(x) | | | | | ADSB_HIGH_RESOLUTION_RF_OUT_PARAMETERS(x) | | | |
|--|---|---|---|---|---|-------------------------------|-------------------------------|-------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| sil | Probability of exceeding the R _c integrity containment radius without detection: unknown | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻³ per flight hour or per operation | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻⁵ per flight hour or per operation | Probability of exceeding the R _c integrity containment radius without detection: 1 × 10 ⁻⁷ per flight hour or per operation | 0 | 1 | 2 | 3 |
| gs | 1 023 | 1 | 2 047 | 0 | 1 022 | 0 | 3 069 | unknown |
| lon8 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| lat8 | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode | As encoded in test CPR_Encode |
| gt | 290 | 0 | 2 042 | 0 | 51 degrees | 0 degrees (north) | 359 degrees | 0 degrees (north) |
| tind | Unknown | Left | Right | Straight | 0 | 1 | 2 | 3 |

Table 7.25 are used in the test "ADSB_High_Resolution_Rec_A".

Table 7.25: Values for use in the test "ADSB_High_Resolution_Rec_A"

| ADSB_HIGH_RESOLUTION_RF_IN_PARAMETERS(x) | | | | | ADSB_HIGH_RESOLUTION_APOUT_PARAMETERS(x) | | | |
|--|-------------------------------|-------------------------------|-------------------------------|-------------------------------|--|--|--|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_B | add_B | add_B | add_B | add_B | add_B | add_B | add_B |
| sil | 0 | 1 | 2 | 3 | Probability of exceeding the R_c integrity containment radius without detection: unknown | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-3} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-5} per flight hour or per operation | Probability of exceeding the R_c integrity containment radius without detection: 1×10^{-7} per flight hour or per operation |
| gs | 1 022 | 0 | 3 069 | unknown | 1 023 | 1 | 2 047 | 0 |
| lon8 | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode |
| lat8 | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode | As encoded in test CPR_Decode |
| gt | 51 degrees | 0 degrees (north) | 359 degrees | 0 degrees (north) | 290 | 0 | 2 042 | 0 |
| tind | 0 | 1 | 2 | 3 | Unknown | Left | Right | Straight |

The values in table 7.26 are used in the test "TIS_Manage_Rec_A".

Table 7.26: Values for use in the test "TIS_Manage_Rec_A"

| TIS_MANAGE_RF_IN_PARAMETERS(x) | | | | | TIS_MANAGE_APPOUT_PARAMETERS(x) | | | |
|---|----------------------------|----------------|----------------|-----------------|--|--|--|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| S | add_A | add_A | add_A | add_A | add_A | add_A | add_A | add_A |
| TIS-B message ID | 0 | 0 | 0 | 0 | Management message | Management message | Management message | Management message |
| TIS-B service version | 0 | 0 | 0 | 0 | TIS-B as defined in the present document | TIS-B as defined in the present document | TIS-B as defined in the present document | TIS-B as defined in the present document |
| TIV ID | 1 | 0 | 3 | 2 | 2 | 1 | 4 | 3 |
| Update period (s) | 11 | 1 | 61 | 28 | 10,0 seconds | 0,5 seconds | 60,0 seconds | 27 seconds |
| Accuracy of TIS-B targets (nmi unless otherwise stated) | 6 | 9 | 1 | 0 | < 0,1 nmi | < 3m | < 10 nmi | unknown |
| Reference point (latitude) | 3 082 | 3 082 | 3 082 | -3 082 | 51 degrees 22 minutes north | 51 degrees 22 minutes north | 51 degrees 22 minutes north | 51 degrees 22 minutes south |
| Reference point (longitude) | -41 | -41 | -41 | -41 | 0 degrees 41 minutes west | 0 degrees 41 minutes west | 0 degrees 41 minutes west | 0 degrees 41 minutes west |
| Lower barometric altitude (ft) | 80 | 0 | 254 | 0 | 18 500 ft | -1 500 ft | 63 500 ft | -1 500 ft |
| Upper barometric altitude (ft) | 160 | 0 | 254 | 0 | 38 500 ft | -1 500 ft | 63 500 ft | -1 500 ft |
| Number of vertices | 1 | 1 | 0 | 1 | 3 vertices | 3 vertices | Circle with centre at reference position and radius specified by one vertex position | 3 vertices |
| TIV1 vertex latitude | 1 | 1 | 1 | 1 | -3 degrees 56 minutes | -3 degrees 56 minutes | -3 degrees 56 minutes | -3 degrees 56 minutes |
| TIV1 vertex longitude | 212 | 212 | 212 | 212 | 2 degrees 11 minutes | 2 degrees 11 minutes | 2 degrees 11 minutes | 2 degrees 11 minutes |
| TIV2 vertex latitude | 64 | 64 | | 64 | -0 degrees 41 minutes | -0 degrees 41 minutes | NOT TRANSMITTED | -0 degrees 41 minutes |
| TIV2 vertex longitude | 31 | 31 | | 31 | -3 degrees 1 minute | -3 degrees 1 minute | NOT TRANSMITTED | -3 degrees 1 minute |
| TIV3 vertex latitude | 220 | 220 | | 220 | 2 degrees 2 minutes | 2 degrees 2 minutes | NOT TRANSMITTED | 2 degrees 2 minutes |
| TIV3 vertex longitude | 130 | 130 | | 130 | 0 degrees 5 minutes | 0 degrees 5 minutes | NOT TRANSMITTED | 0 degrees 5 minutes |

The values in table 7.27 are used in the test "TIS_Air_Air_Rec_A".

Table 7.27: Values for use in the test "TIS_Air_Air_Rec_A"

| TIS_AIR_AIR_RF_IN_PARAMETERS(x) | | | | | TIS_AIR_AIR_APPOUT_PARAMETERS(x) | | | |
|---------------------------------|--|--|--|-----------------|--|--|--|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| S | add_A | add_A | add_A | add_A | add_A | add_A | add_A | add_A |
| TIS-B message ID | 1 | 1 | 1 | 1 | Aircraft target (airborne service) message | Aircraft target (airborne service) message | Aircraft target (airborne service) message | Aircraft target (airborne service) message |
| TIV ID | 1 | 0 | 3 | 2 | 2 | 1 | 4 | 3 |
| Target Identifier | 0000 10 101 010 00 00 0000 0001 | 0000 10 101 010 00 00 0000 0010 | 0000 10 101 010 00 00 0000 0011 | 1200 | 0000 10 101 010 00 00 0000 0001 | 0000 10 101 010 00 00 0000 0010 | 0000 10 101 010 00 00 0000 0011 | 1200 |
| Target Identifier flag | 0 | 0 | 0 | 1 | Aircraft address | Aircraft address | Aircraft address | Special Identifier |
| Radar fusion flag | 0 | 1 | 0 | 1 | TIS-B target based only on radar or multilat. data | TIS-B target includes ADB-B as a data source | TIS-B target based only on radar or multilat. data | TIS-B target includes ADB-B as a data source |
| ADS-B fault flag | | 0 | | 1 | NOT TRANSMITTED | ADS-B corresponds with TIS-B | NOT TRANSMITTED | ADS-B diverts from TIS-B |
| Latitude | -11 036 | -11 036 | -11 036 | -11 036 | 50,0166 degrees north | 50,0166 degrees north | 50,0166 degrees north | 50,0166 degrees north |
| Longitude | -13 173 | -13 173 | -13 173 | -13 173 | 1,8499 degrees west | 1,8499 degrees west | 1,8499 degrees west | 1,8499 degrees west |
| Barometric altitude | 532 | 2 | 4 072 | 132 | 4 000 feet | -1 300 feet | 130 000 feet | 0 feet |
| Ground speed | 1 023 | 1 | 2 047 | 0 | 1 022 knots | 0 knots | 3 069 knots | unknown |
| Ground track | 31 | 0 | 255 | 0 | 44,00 degrees | 0,00 degrees | 359,70 degrees | 0,00 degrees |
| Time Stamp | 20 | 0 | 59 | 0 | 4 000 ms after time reference | time reference | 11,8 seconds after time reference | time reference |
| Flight ID Flag | 1 | 1 | 1 | 0 | optional information is present | optional information is present | optional information is present | optional information is not present |
| Flight ID type (optional) | 0 | 1 | 1 | | callsign | registration marking (tail number) | registration marking (tail number) | |
| Aircraft category (optional) | 9 | 5 | 10 | | rotorcraft | highly manoeuvrable and high speed | glider/sailplane | |

The values in table 7.28 are used in the test "TIS_Air_Ground_Rec_A".

Table 7.28: Values for use in the test "TIS_Air_Ground_Rec_A"

| TIS_AIR_GRD_RF_IN_PARAMETERS(x) | | | | | TIS_AIR_GRD_APPOUT_PARAMETERS(x) | | | |
|---------------------------------|--|--|--|-----------------|--|--|--|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| S | add_A | add_A | add_A | add_A | add_A | add_A | add_A | add_A |
| TIS-B message ID | 2 | 2 | 2 | 2 | Aircraft target (ground service) message | Aircraft target (ground service) message | Aircraft target (ground service) message | Aircraft target (ground service) message |
| TIV ID | 1 | 0 | 3 | 2 | 2 | 1 | 4 | 3 |
| Target Identifier | 0000 10 101 010 00 00 0000 0001 | 0000 10 101 010 00 00 0000 0010 | 0000 10 101 010 00 00 0000 0011 | 1200 | 0000 10 101 010 00 00 0000 0001 | 0000 10 101 010 00 00 0000 0010 | 0000 10 101 010 00 00 0000 0011 | 1200 |
| Target Identifier flag | 0 | 0 | 0 | 1 | Aircraft address | Aircraft address | Aircraft address | Special Identifier |
| Radar fusion flag | 0 | 1 | 0 | 1 | TIS-B target based only on radar or multilat. data | TIS-B target includes ADB-B as a data source | TIS-B target based only on radar or multilat. data | TIS-B target includes ADB-B as a data source |
| ADS-B fault flag | | 0 | | 1 | NOT TRANSMITTED | ADS-B corresponds with TIS-B | NOT TRANSMITTED | ADS-B diverts from TIS-B |
| Latitude | -215 820 | -215 820 | -215 820 | -215 820 | 50,0166 degrees north | 50,0166 degrees north | 50,0166 degrees north | 50,0166 degrees north |
| Longitude | -210 773 | -210 773 | -210 773 | -210 773 | 1,8499 degrees west | 1,8499 degrees west | 1,8499 degrees west | 1,8499 degrees west |
| Ground speed | 1 023 | 1 | 2 047 | 0 | 1 022 knots | 0 knots | 3 069 knots | unknown |
| Ground track | 31 | 0 | 255 | 0 | 44,00 degrees | 0,00 degrees | 359,70 degrees | 0,00 degrees |
| Time Stamp | 20 | 0 | 59 | 0 | 4 000 ms after time reference | time reference | 11,8 seconds after time reference | time reference |
| Flight ID Flag | 1 | 1 | 1 | 0 | optional information is present | optional information is present | optional information is present | optional information is not present |
| Flight ID type (optional) | 0 | 1 | 1 | | callsign | registration marking (tail number) | registration marking (tail number) | |
| Aircraft category (optional) | 9 | 5 | 10 | | rotorcraft | highly manoeuvrable and high speed | glider/sailplane | |

The values in table 7.29 are used in the test "TIS_Ground_Veh_Rec_A".

Table 7.29: Values for use in the test "TIS_Ground_Veh_Rec_A"

| TIS_GRD_VEH_RF_IN_PARAMETERS(x) | | | | | TIS_GRD_VEH_APOUT_PARAMETERS(x) | | | |
|---------------------------------|--|--|--|-----------------|--|--|--|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| S | add_A | add_A | add_A | add_A | add_A | add_A | add_A | add_A |
| TIS-B message ID | 3 | 3 | 3 | 3 | Ground Vehicle target (ground service) message | Ground Vehicle target (ground service) message | Ground Vehicle target (ground service) message | Ground Vehicle target (ground service) message |
| TIV ID | 1 | 0 | 3 | 2 | 2 | 1 | 4 | 3 |
| Target Identifier | 0000 10 101 010 00 00 0000 0001 | 0000 10 101 010 00 00 0000 0010 | 0000 10 101 010 00 00 0000 0011 | 1200 | 0000 10 101 010 00 00 0000 0001 | 0000 10 101 010 00 00 0000 0010 | 0000 10 101 010 00 00 0000 0011 | 1200 |
| Target Identifier flag | 0 | 0 | 0 | 1 | Aircraft address | Aircraft address | Aircraft address | Special Identifier |
| Radar fusion flag | 0 | 1 | 0 | 1 | TIS-B target based only on radar or multilat. data | TIS-B target includes ADB-B as a data source | TIS-B target based only on radar or multilat. data | TIS-B target includes ADB-B as a data source |
| ADS-B fault flag | | 0 | | 1 | NOT TRANSMITTED | ADS-B corresponds with TIS-B | NOT TRANSMITTED | ADS-B diverts from TIS-B |
| Latitude | -215 820 | -215 820 | -215 820 | -215 820 | 50,0166 degrees north | 50,0166 degrees north | 50,0166 degrees north | 50,0166 degrees north |
| Longitude | -210 773 | -210 773 | -210 773 | -210 773 | 1,8499 degrees west | 1,8499 degrees west | 1,8499 degrees west | 1,8499 degrees west |
| Ground speed | 1 023 | 1 | 127 | 0 | 1 022 knots | 0 knots | 126 knots | unknown |
| Ground track | 31 | 0 | 255 | 0 | 44,00 degrees | 0,00 degrees | 359,70 degrees | 0,00 degrees |
| Time Stamp | 20 | 0 | 59 | 0 | 4 000 ms after time reference | time reference | 11,8 seconds after time reference | time reference |

The values in table 7.30 are used in the test "FIS_Metar_Rec_A".

Table 7.30: Values for use in the test "FIS_Metar_Rec_A"

| FIS_METAR_RF_IN_PARAMETERS(x) | | | | | FIS_METAR_APOUT_PARAMETERS(x) | | | |
|-------------------------------|----------------------------|----------------|----------------|-----------------|---|---|---|---|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| S | add_G | add_G | add_G | add_G | add_G | add_G | add_G | add_G |
| Fmi | 0 | 0 | 0 | 0 | METAR message | METAR message | METAR message | METAR message |
| Vers | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Med | 3 | 0 | 7 | 0 | 3 | 0 | 7 | 0 |
| Locl | 05 07 12 12 | 12 06 16 07 | 12 05 02 12 | 05 04 04 06 | EGLL | LFPG | LEBL | EDDF |
| Date | 14 | 1 | 31 | 0 | day 14 | day 1 | day 31 | unknown |
| Ti | 699 | 1 | 1 440 | 1 | 698 minutes | 0 minutes | 1 439 minutes | 0 minutes |
| Wdi | 15 | 1 | 36 | 1 | 140 degrees | 0 degrees | 350 degrees | 0 minutes |
| Wsp | 46 | 1 | 102 | 1 | 45 kts | 0 kts | 100,5 kts | 0 kts |
| ftxt | 1 | 0 | 0 | 1 | Message | None | None | Message |
| Mxw | 1 | 0 | 1 | 0 | gusts present | None | gusts present | None |
| Cav | 1 | 0 | 1 | 0 | No CAVOK | No CAVOK | No CAVOK | CAVOK |
| Trd | 1 | 0 | 1 | 0 | TREND report present | No TREND report | TREND report present | No TREND report |
| Pwf | 1 | 0 | 2 | | 2 weather fields included | 1 weather Field included | 3 weather Field included | NOT TRANSMITTED |
| Rwf | 1 | 0 | 2 | 0 | 2 weather fields included | 1 weather Field included | 3 weather fields included | 1 weather Field included |
| Qfef | 0 | 0 | 0 | 1 | Present | Present | Present | Not present |
| Temp | 100 | 1 | 140 | 80 | +20°C | -7°C | +6°C | 0°C |
| Dew | 105 | 1 | 140 | 80 | +2°C | -7°C | +6°C | 0°C |
| Qnh | 153 | 1 | 251 | 0 | 1 002,5 | 850,5 | 1 100,5 | Unknown |
| Ssf | 1 | 0 | 1 | 0 | State of the sea and sea temperature fields not present | State of the sea and sea temperature fields present | State of the sea and sea temperature fields not present | State of the sea and sea temperature fields present |
| Vdit | 16 | 1 | 36 | 1 | 150 degrees | 0 degrees | 350 degrees | 0 degrees |
| Vdif | 16 | 1 | 36 | 1 | 150 degrees | 0 degrees | 350 degrees | 0 degrees |
| Wgi | 47 | 1 | 102 | 1 | 46,0 kts | 0,0 kts | 100,5 kts | 0,0 kts |
| Mis | 16 | 1 | 45 | | 800 meters | 0 meters | 10 000 meters | NOT TRANSMITTED |
| misd | 0 | 1 | 5 | | North | North East | South West | NOT TRANSMITTED |
| Mas | 18 | 1 | 45 | | 1 000 meters | 0 meters | 10 000 meters | NOT TRANSMITTED |
| masd | 7 | 4 | 2 | | North West | South | East | NOT TRANSMITTED |
| Pw1 | 18 | 10 | 18 | | No precipitation during observation, funnel clouds | No precipitation during observation, brune-mist | No precipitation during observation, funnel clouds | NOT TRANSMITTED |

| FIS_METAR_RF_IN_PARAMETERS(x) | | | | | FIS_METAR_APPOUT_PARAMETERS(x) | | | |
|--------------------------------------|----------------------------|----------------|------------------|----------------------|---|---|--|---|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| Pw2 | 64 | | 40 | | precipitation during observation, heavy rain | NOT TRANSMITTED | No precipitation during observation, fog patches | NOT TRANSMITTED |
| Pw3 | | | 87 | | NOT TRANSMITTED | NOT TRANSMITTED | precipitation during observation, soft hail | NOT TRANSMITTED |
| Cno | 0 | 2 | 3 | | 1 cloud field to follow | 3 cloud fields to follow | 4 cloud fields to follow | NOT TRANSMITTED |
| Ch1 | 98 | 0 | 156 | | 2 970 meters | 0 meters | 20 100 meters | NOT TRANSMITTED |
| Ch2 | | 2 | 4 | | NOT TRANSMITTED | 90 meters | 150 meters | NOT TRANSMITTED |
| Ch3 | | 50 | 155 | | NOT TRANSMITTED | 1 530 meters | 19 800 meters | NOT TRANSMITTED |
| Ch4 | | | 96 | | NOT TRANSMITTED | NOT TRANSMITTED | 2 910 meters | NOT TRANSMITTED |
| Cc1 | 4 | 1 | 4 | | Overcast | Few | overcast | NOT TRANSMITTED |
| Cc2 | | 2 | 1 | | NOT TRANSMITTED | scattered | few | NOT TRANSMITTED |
| Cc3 | | 3 | 5 | | NOT TRANSMITTED | broken | sky clear | NOT TRANSMITTED |
| Cc4 | | | 1 | | NOT TRANSMITTED | NOT TRANSMITTED | few | NOT TRANSMITTED |
| Cty1 | 2 | 0 | 2 | | towering cumulus | No clouds of concern | towering cumulus | NOT TRANSMITTED |
| Cty2 | | 1 | 0 | | NOT TRANSMITTED | cumulonimbus | No clouds of concern | NOT TRANSMITTED |
| Cty3 | | 2 | 1 | | NOT TRANSMITTED | towering cumulus | cumulonimbus | NOT TRANSMITTED |
| Cty4 | | | 0 | | NOT TRANSMITTED | NOT TRANSMITTED | No clouds of concern | NOT TRANSMITTED |
| Qfe | 151 | 1 | 251 | | 1 000,5 | 850,5 | 1 100,5 | NOT TRANSMITTED |
| Rw1 | 68 | 05 | 97 | 18 | Precipitation during observation, rain and snow | No precipitation during observation, dust haze | Precipitation during observation, heavy thunderstorm | No precipitation during observation, Funnel Cloud |
| Rw2 | 56 | | 83 | | Precipitation during observation, freezing rain | NOT TRANSMITTED | Precipitation during observation, showers of rain and snow | NOT TRANSMITTED |
| Rw3 | | | 79 | | NOT TRANSMITTED | NOT TRANSMITTED | Precipitation during observation, ice pellets | NOT TRANSMITTED |
| Sst | 35 | 1 | 51 | 11 | +24,5°C | -9,5°C | +40°C | 0°C |
| Ss | 4 | 1 | 10 | 0 | 3 | 0 | 9 | Unknown |
| Txt (TREND) TXT (FREE TEXT) | TREND MESSAGE | | TREND MESSAGE | TEST FREE TEXT | TREND MESSAGE | NOT TRANSMITTED | TREND MESSAGE | NOT TRANSMITTED |
| | \$ TEST FREE TEXT | | \$ | | TEST FREE TEXT | TEST FREE TEXT | TEST FREE TEXT | TEST FREE TEXT |

The values in table 7.31 are used in the test "FIS_ATIS_Rec_A".

Table 7.31: Values for use in the test "FIS_ATIS_Rec_A"

| FIS_ATIS_RF_IN_PARAMETERS(x) | | | | | FIS_ATIS_APPOUT_PARAMETERS(x) | | | |
|------------------------------|----------------------------|----------------|----------------|-----------------|-------------------------------|--------------------------|---------------------------|--------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q | Set R | Set S |
| S | add_G | add_G | add_G | add_G | Add_G | add_G | add_G | Add_G |
| Fmi | 1 | 1 | 1 | 1 | ATIS message | ATIS message | ATIS message | ATIS message |
| Vers | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Med | 3 | 0 | 7 | 0 | 3 | 0 | 7 | 0 |
| Locl | 05 07 12 12 | 12 06 16 07 | 12 05 02 12 | 05 04 04 06 | EGLL | LFPG | LEBL | EDDF |
| Date | 17 | 1 | 31 | 0 | day 17 | day 1 | day 31 | 0 |
| Ti | 633 | 2 | 1 440 | 1 | 632 minutes | 1 minute | 1 439 minutes | 0 minutes |
| Hd | 6 | 0 | 12 | 0 | 30 minutes | 0 minutes | 60 minutes | 0 minutes |
| trans | 21 | 1 | 61 | 1 | 100 | 0 | 300 | 0 |
| Oth | | | | | | | | |
| Aavl (bit 12) | 1 | 0 | 1 | 1 | Open | Closed | Open | Open |
| Aavl (bits 1-11) | 257 | 1 | 1 440 | 1 | 256 | 0 | 1 439 | 0 |
| Wdi | 15 | 1 | 36 | 1 | 140 degrees | 0 degrees | 350 degrees | 0 degrees |
| Wsp | 46 | 1 | 102 | 1 | 45 kts | 0 kts | 100,5 kts | 0 kts |
| Tem | 98 | 1 | 140 | 80 | +18°C | -79°C | +60°C | 0 °C |
| Dew | 103 | 1 | 140 | 80 | +23°C | -79°C | +60°C | 0°C |
| Qnh | 153 | 1 | 251 | 0 | 1 002,5 | 850,5 | 1 100,5 | Unknown |
| Ruf | 1 | 0 | 3 | 1 | 3 runway fields | 1 runway field | 4 runway fields | 2 Runway fields |
| Ftxt | 1 | 1 | 1 | 0 | message | message | message | None |
| mxw | 1 | 0 | 1 | 0 | gusts present | none | gusts present | None |
| Cav | 0 | 0 | 0 | 1 | No CAVOK | No CAVOK | No CAVOK | CAVOK |
| Trd | 1 | 0 | 1 | 1 | Message | None | Message | Message |
| Qfef | 0 | 0 | 1 | 0 | Present | Present | Not Present | Present |
| Ws1 | 1 | 1 | 1 | 0 | Yes | Yes | Yes | No |
| Ws2 | 1 | 1 | 1 | 0 | yes | NOT TRANSMITTED | yes | no |
| Ws3 | 0 | | 1 | | no | NOT TRANSMITTED | yes | NOT TRANSMITTED |
| Ws4 | | | 1 | | NOT TRANSMITTED | NOT TRANSMITTED | yes | NOT TRANSMITTED |
| Pwf | 1 | 0 | 2 | | 2 weather fields included | 1 weather Field included | 3 weather fields included | NOT TRANSMITTED |
| Rwf | 2 | 0 | 2 | 1 | 3 weather fields included | 1 weather Field included | 3 weather fields included | 2 weather Field included |
| Vdit | 18 | 1 | 36 | 0 | 170 degrees | 0 degrees | 350 degrees | Unknown |
| Vdif | 16 | 1 | 36 | 0 | 150 degrees | 0 degrees | 350 degrees | Unknown |
| Wgi | 47 | 1 | 102 | 1 | 46 kts | 0,1 kts | 100,5 kts | 0 kts |
| Mis | 15 | 1 | 45 | | 775 meters | 50 meters | 10 500 meters | NOT TRANSMITTED |
| Misd | 3 | 0 | 5 | | South East | North | South West | NOT TRANSMITTED |
| mas | 41 | 1 | 45 | | 6 500 meters | 50 meters | 10 500 meters | NOT TRANSMITTED |

| FIS_ATIS_RF_IN_PARAMETERS(x) | | | | | FIS_ATIS_APPOUT_PARAMETERS(x) | | | |
|------------------------------|----------------------------|----------------|------------------|----------------------|---|---|--|---|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q | Set R | Set S |
| masd | 5 | 1 | 6 | | South West | North East | West | NOT TRANSMITTED |
| Cno | 1 | 0 | 3 | | 2 cloud fields to follow | 1 cloud field to follow | 4 cloud fields to follow | NOT TRANSMITTED |
| Cc1 | 4 | 1 | 2 | | Broken | Few | Scattered | NOT TRANSMITTED |
| Cc2 | 3 | | 3 | | Overcast | NOT TRANSMITTED | Broken | NOT TRANSMITTED |
| Cc3 | | | 4 | | NOT TRANSMITTED | NOT TRANSMITTED | Overcast | NOT TRANSMITTED |
| Cc4 | | | 5 | | NOT TRANSMITTED | NOT TRANSMITTED | Sky clear | NOT TRANSMITTED |
| Ch1 | 98 | 0 | 156 | | 2 980 meters | 30 meters | 20 300 meters | NOT TRANSMITTED |
| Ch2 | 98 | | 100 | | 2 980 meters | NOT TRANSMITTED | 3 300 meters | NOT TRANSMITTED |
| Ch3 | | | 97 | | NOT TRANSMITTED | NOT TRANSMITTED | 2 960 meters | NOT TRANSMITTED |
| Ch4 | | | 156 | | NOT TRANSMITTED | NOT TRANSMITTED | 20 300 meters | NOT TRANSMITTED |
| Cty1 | 1 | 0 | 2 | | Cumulonimbus | No clouds of concern | Towering Cumulus | NOT TRANSMITTED |
| Cty2 | 2 | | 1 | | Towering Cumulus | NOT TRANSMITTED | Cumulonimbus | NOT TRANSMITTED |
| Cty3 | | | 0 | | NOT TRANSMITTED | NOT TRANSMITTED | No clouds of concern | NOT TRANSMITTED |
| Cty4 | | | 2 | | NOT TRANSMITTED | NOT TRANSMITTED | Towering Cumulus | NOT TRANSMITTED |
| Pw1 | 38 | 04 | 97 | | No precipitation during observation, Blowing snow | No precipitation during observation, Fume-smoke | precipitation during observation, Heavy thunderstorms | NOT TRANSMITTED |
| pw2 | 04 | | 20 | | No precipitation during observation, Fume-smoke | NOT TRANSMITTED | No precipitation during observation, Recent drizzle | NOT TRANSMITTED |
| pw3 | | | 33 | | NOT TRANSMITTED | NOT TRANSMITTED | No precipitation during observation, heavy sand storm | NOT TRANSMITTED |
| rw1 | 68 | 05 | 97 | 18 | precipitation during observation, Rain and snow | no precipitation during observation, Dust haze | precipitation during observation, Heavy thunderstorm | No precipitation during observation, funnel cloud |
| rw2 | 58 | | 38 | 38 | precipitation during observation, rain | NOT TRANSMITTED | no precipitation during observation, Blowing snow | no precipitation during observation, Blowing snow |
| rw3 | 87 | | 57 | | precipitation during observation, soft hail | NOT TRANSMITTED | precipitation during observation, heavy freezing drizzle | NOT TRANSMITTED |
| qfe | 151 | 1 | | 0 | 1 000,5 | 850,5 | NOT TRANSMITTED | Unknown |
| Txt (TREND) | TREND MESSGAE | | TREND MESSGAE | TEST FREE TEXT | TREND MESSGAE | NOT TRANSMITTED | TREND MESSGAE | NOT TRANSMITTED |
| Txt (FREE TEXT) | \$ TEST FREE TEXT | | \$ | | TEST FREE TEXT | NOT TRANSMITTED | NOT TRANSMITTED | TEST FREE TEXT |

The values in table 7.32 are used in the test "FIS_RCN_Rec_A".

Table 7.32: Values for use in the test "FIS_RCN_Rec_A"

| FIS_RCN_RF_IN_PARAMETERS(x) | | | | | FIS_RCN_APPOUT_PARAMETERS(x) | | | |
|-----------------------------|----------------------------|----------------|----------------|-----------------|-------------------------------|------------------------------|-----------------------------------|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q | Set R | Set S |
| S | add_G | add_G | add_G | add_G | Add_G | add_G | add_G | Add_G |
| Fmi | 1 | 1 | 1 | 1 | RCM message | RCM message | RCM message | RCM message |
| Vers | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Med | 3 | 0 | 7 | 0 | 3 | 0 | 7 | 0 |
| Locl | 05 07 12 12 | 12 06 16 07 | 12 05 02 12 | 05 04 04 06 | EGLL | LFPG | LEBL | EDDF |
| Date | 17 | 1 | 31 | 0 | day 17 | day 1 | day 31 | 0 |
| Ti | 633 | 2 | 1 440 | 1 | 632 minutes | 1 minute | 1 439 minutes | 0 minutes |
| brk | 3 | 5 | 4 | 6 | Braking action: medium | Braking action: good | Braking action: medium/good | Figures unreliable |
| rdp | 1 | 0 | 6 | 0 | Damp | Clear and dry | Slush | Clear and dry |
| rcon | 3 | 0 | 4 | 4 | 30 % | None | 100 % | 53 % |
| dod | 15 | 0 | 97 | 98 | 15 mm | 0 mm | 40 cm | Runway not operational due to snow, ice, large drifts or runway clearance and depth not reported |
| Run (bits 7 to 8) | 2 | 0 | 0 | 0 | Right runway | single runway | single runway | single runway |
| Run (bits 1 to 7) | 13 | 0 | 23 | 29 | 14 | 01 | 24 | 30 |
| Aptf | 1 | 0 | 2 | 0 | 2 approach fields included | 1 approach field included | 3 approach fields included | 1 approach field included |
| Apty1 | 1 | 0 | 9 | 8 | ILS CAT II | ILS CAT I | Visual | MLS |
| Apty2 | 3 | | 8 | | VOR | NOT TRANSMITTED | MLS | NOT TRANSMITTED |
| Apty3 | | | 7 | | NOT TRANSMITTED | NOT TRANSMITTED | VDF | NOT TRANSMITTED |
| rrl | 50 | 0 | 255 | 0 | 2 475 meters | 0 meters | Full length | 0 meters |
| rrw | 33 | 1 | 127 | 1 | 32 meters | 0 meters | Full width | 0 meters |
| rdp | 1 | 1 | 0 | 0 | Negative | Negative | Positive | Positive |
| rvt | 23 | 1 | 32 | | 700 meters | 0 meters | 1 500 meters | NOT TRANSMITTED |
| rvm | 22 | 1 | 32 | | 650 meters | 0 meters | 1 527 meters | NOT TRANSMITTED |
| rve | 24 | 1 | 32 | | 750 meters | 0 meters | 1 500 meters | NOT TRANSMITTED |
| rso | 1 | 0 | 2 | 1 | Human Observer | Unknown | Instrumented RVR system (IRVR) | Human Observer |
| rus | 0 | 2 | 2 | 2 | Yes - for arrivals | Yes - mixed operations | Yes - mixed operations | Yes - mixed operations |
| Ravl (bit 12) | 1 | 0 | 1 | 0 | Open | Closed | Open | Closed |
| Ravl (bits 1 to 11) | 257 | 1 | 124 | 0 | 256 | 0 | 123 | Unknown |
| Ws | 1 | 1 | 1 | 0 | Yes | Yes | Yes | No |

The values in table 7.33 are used in the test "FIS_SIGMET_REC_A".

Table 7.33: Values for use in the test "FIS_SIGMET_REC_A"

| FIS_SIGMET_RF_IN_PARAMETERS(x) | | | | | FIS_SIGMET_APPOUT_PARAMETERS(x) | | | |
|--------------------------------|----------------------------|-------------------|-------------------|-----------------|---------------------------------|-----------------------------|------------------------------------|-------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| S | add_G | add_G | add_G | add_G | add_G | add_G | add_G | add_G |
| Fmi | 3 | 3 | 3 | 3 | SIGMET message | SIGMET message | SIGMET message | SIGMET message |
| Sqn | 6 | 0 | 15 | 4 | 7 | 1 | 16 | 5 |
| med | 2 | 0 | 7 | 3 | C | A | H | D |
| Locl | 05 07 12 12 | 12 06 06 07 | 12 05 02 12 | 05 04 04 06 | EGLL | LFPG | LEBL | EDDF |
| Ori | 05 07 12 12 | 12 06 06 07 | 12 05 02 12 | 05 04 04 06 | EGLL | LFPG | LEBL | EDDF |
| Di | 14 | 1 | 31 | 0 | day 14 | day 1 | day 31 | unknown |
| Ti | 699 | 1 | 1 440 | 1 | 698 minutes | 0 minutes | 1 439 minutes | 0 minutes |
| Di | 14 | 1 | 31 | 0 | day 14 | day 1 | day 31 | unknown |
| Ti | 721 | 2 | 1 440 | 1 | 720 minutes | 1 minutes | 1 439 minutes | 0 minutes |
| Ftxt | 1 | 1 | 1 | 0 | Message | Message | Message | None |
| Sfl | 0 | 1 | 0 | 1 | Normal | Convective SIGMET | Normal | Convective SIGMET |
| crp | 17 | 5 | 0 | 13 | Heavy dust storm | Thunderstorm squall line | Condition reported in free text | Severe turbulence |
| txt | test free text | test free text | test free text | | test free text | test free text | test free text | NOT TRANSMITTED |

The values in table 7.34 are used in the test "FIS_TSA_CTR_REC_A".

Table 7.34: Values for use in the test "FIS_TSA_CTR_REC_A"

| FIS_TSA_CTR_RF_IN_PARAMETERS(x) | | | | | FIS_TSA_CTR_APPOUTPARAMETERS(x) | | | |
|---------------------------------|----------------------------|----------------|----------------|-----------------|---------------------------------|----------------|----------------|-----------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| S | add_G | add_G | add_G | add_G | add_G | add_G | add_G | add_G |
| Fmi | 6 | 6 | 6 | 6 | TSA message | TSA message | TSA message | TSA message |
| Vers | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Med | 14 | 1 | 31 | 0 | 2 | 0 | 7 | 3 |
| Date | 14 | 1 | 31 | 0 | day 14 | day 1 | day 31 | unknown |
| Time | 699 | 1 | 1 440 | 1 | 698 minutes | 0 minutes | 1 439 minutes | 0 minutes |
| NoT | 0 | 0 | 0 | 0 | 1 TSA | 1 TSA | 1 TSA | 1 TSA |
| Toa | 0 | 0 | 0 | 0 | Control Zone | Control Zone | Control Zone | Control Zone |
| Act | 0 | 1 | 1 | 1 | TSA Not Active | TSA Active | TSA Active | TSA Active |
| Locl | 05 07 12 12 | 12 06 16 07 | 12 05 02 12 | 05 04 04 06 | EGLL | LFPG | LEBL | EDDF |
| Sloc | 1 | 2 | 3 | 4 | A | B | C | D |
| Actdat | 14 | 2 | 31 | 5 | day 14 | day 2 | day 1 | day 5 |
| Actim | 703 | 6 | 6 | 56 | 702 minutes | 5 minutes | 5 minutes | 55 minutes |
| freq | 81 | 1 | 1 160 | 0 | 110,025 | 108,000 | 136,975 | unknown |

The values in table 7.35 are used in the test "FIS_TSA_DNG_REC_A".

Table 7.35: Values for use in the test "FIS_TSA_DNG_REC_A"

| FIS_TSA_DNG_RF_IN_PARAMETERS(x) | | | | | FIS_TSA_DNG_APPOUTPARAMETERS(x) | | | |
|---------------------------------|----------------------------|----------------|----------------|-----------------|---------------------------------|---------------------|-----------------|--------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| S | add_G | add_G | add_G | add_G | add_G | add_G | add_G | add_G |
| Fmi | 6 | 6 | 6 | 6 | TSA message | TSA message | TSA message | TSA message |
| Vers | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Med | 14 | 1 | 31 | 0 | 2 | 0 | 7 | 3 |
| Date | 14 | 1 | 31 | 0 | day 14 | day 1 | day 31 | unknown |
| Time | 699 | 1 | 1 440 | 1 | 698 minutes | 0 minutes | 1 439 minutes | 0 minutes |
| NoT | 0 | 0 | 0 | 0 | 1 TSA | 1 TSA | 1 TSA | 1 TSA |
| Toa | 2 | 1 | 3 | 2 | Danger Zone | Restricted Zone | Prohibited Zone | Danger Zone |
| Act | 0 | 1 | 1 | 1 | TSA Not Active | TSA Active | TSA Active | TSA Active |
| Stid | 05 19 | 05 07 | 12 18 | 12 15 | Sweden (ES) | United Kingdom (EG) | Romania (LR) | Liechtenstein (LO) |
| Aln | 43 | 0 | 8 171 | 0 | 44 | 1 | 8 192 | 1 |
| Saln | 4 | 3 | 2 | 1 | D | C | B | A |
| Actdat | 14 | 2 | 31 | 5 | day 14 | day 2 | day 1 | day 5 |
| Actim | 703 | 6 | 6 | 56 | 702 minutes | 5 minutes | 5 minutes | 55 minutes |

The values in table 7.36 are used in the test "FIS_REQUEST_A".

Table 7.36: Values for use in the test "FIS_REQUEST_A"

| FIS_REQUEST_APPIN_PARAMETERS(x) | | | | | FIS_REQUEST_RF_OUT_PARAMETERS(x) | | | |
|---------------------------------|-------------------------|----------------|----------------|-----------------|----------------------------------|----------------|----------------|-----------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_A | add_A | add_A | add_A | add_A | add_A | add_A | add_A |
| fmi | REPORT REQUEST | REPORT REQUEST | REPORT REQUEST | REPORT REQUEST | 5 | 5 | 5 | 5 |
| mrq | SIGMET message | METAR message | TSA message | METAR message | 3 | 0 | 6 | 0 |
| locl | EGLL | LFPG | LEBL | EDDF | 05 07 12 12 | 12 06 06 07 | 12 05 02 12 | 05 04 04 06 |

The values in table 7.37 are used in the test "FIS_REQUEST_B".

Table 7.37: Values for use in the test "FIS_REQUEST_B"

| FIS_REQUEST_APPIN_PARAMETERS(x) | | | | | FIS_REQUEST_RF_OUT_PARAMETERS(x) | | | |
|---------------------------------|-------------------------|----------------|----------------|-----------------|----------------------------------|----------------|----------------|-----------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_A | add_A | add_A | add_A | add_A | add_A | add_A | add_A |
| fmi | REPORT REQUEST | REPORT REQUEST | REPORT REQUEST | REPORT REQUEST | 5 | 5 | 5 | 5 |
| mrq | SIGMET message | METAR message | TSA message | METAR message | 3 | 0 | 6 | 0 |
| locl | EGLL | LFPG | LEBL | EDDF | 05 07 12 12 | 12 06 06 07 | 12 05 02 12 | 05 04 04 06 |

The values in table 7.38 are used in the test "GNS_Type1_Message_Rec_A".

Table 7.38: Values for use in the test "GNS_Type1_Message_Rec_A"

| GNS_TYPE1_RF_IN_PARAMETERS(x) | | | | | GNS_TYPE1_APOUT_PARAMETERS(x) | | | |
|-------------------------------|-------------------------|--------------------|--------------------|--------------------|---|--|--|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_G | add_G | add_G | add_G | add_G | add_G | add_G | add_G |
| mbi | 11111 | 10101 | 11111 | 10101 | Test GNS-B message | Normal GNS-B message | Test GNS-B message | Normal GNS-B message |
| gid | 5 7 12 12 | 12 06 06 07 | 12 05 02 12 | 05 04 04 06 | EGLL | LFPG | LEBL | EDDF |
| gmi | 1 | 1 | 1 | 1 | Type 1 message | Type 1 message | Type 1 message | Type 1 message |
| len | 27 | 27 | 27 | 27 | 27 bytes | 27 bytes | 27 bytes | 27 bytes |
| amf | 1 | 0 | 3 | 0 | This is the first of two Type 1 Messages in a frame containing measurement blocks | All measurement blocks are contained in one Type 1 Message | This is the second of two Type 1 Messages in a frame containing measurement blocks | All measurement blocks are contained in one Type 1 Message |
| zc | 600 | 0 | 12 000 | 0 | 01 minute 00,0 seconds | 00 minutes 00,0 seconds | 20 minutes 00,0 seconds | 00 minutes 00,0 seconds |
| n | 1 | 1 | 1 | 1 | 1 block | 1 block | 1 block | 1 block |
| t | 0 | 0 | 0 | 0 | C/A or CSA code L1 | C/A or CSA code L1 | C/A or CSA code L1 | C/A or CSA code L1 |
| edp | 2 | 1 | 255 | 1 | 5×10^{-6} m/m | 0 m/m | $1,270 \times 10^{-3}$ m/m | 0 m/m |
| ec | See clause 5.4.5.3 | See clause 5.4.5.3 | See clause 5.4.5.3 | See clause 5.4.5.3 | See clause 5.4.5.3 | See clause 5.4.5.3 | See clause 5.4.5.3 | See clause 5.4.5.3 |

| GNS_TYPE1_RF_IN_PARAMETERS(x) | | | | | GNS_TYPE1_APOUT_PARAMETERS(x) | | | |
|-------------------------------|-------------------------|------------------|------------------|------------------|---|------------------|---|------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| ad | 1 | 0 | 255 | 0 | 10 seconds | 0 seconds | Prediction of source availability duration not provided | 0 seconds |
| id | 2 | 1 | 138 | 1 | ID2 | ID1 | ID138 | ID1 |
| iod | 1 | 0 | 2 | 0 | GLONASS | GPS | SBAS | GPS |
| prc | 1 | 0 | 65 534 | 32 767 | -327,66 m | -327,67 m | +327,67 m | 0 m |
| rrc | 1 | 0 | 65 534 | 32 767 | -32,766 m | -32,767 m | +32,767 m | 0 m |
| sd | 1 | 0 | 255 | 0 | 0,02 m | 0 m | Ranging source correction invalid | 0 m |
| b1 (bit 8) | 1 | 0 | 1 | 0 | Negative | Positive | Negative | Positive |
| b1 (bits 1 to 7) | 1 | 0 | 127 | 0 | 0,05 m | 0 m | 6,35 m | 0 m |
| b2 (bit 8) | 1 | 0 | 1 | 0 | Negative | Positive | Negative | Positive |
| b2 (bits 1 to 7) | 126 | 0 | 127 | 0 | 6,34 m | 0 m | 6,35 m | 0 m |
| b3 (bit 8) | 1 | 0 | 1 | 0 | Reference receiver was not used to compute the pseudorange correction | Positive | Negative | Positive |
| b3 (bits 1 to 7) | 0 | 0 | 127 | 0 | Reference receiver was not used to compute the pseudorange correction | 0 m | 6,35 m | 0 m |
| b4 (bit 8) | 0 | 0 | 1 | 0 | Positive | Positive | Negative | Positive |
| b4 (bits 1 to 7) | 126 | 0 | 127 | 0 | 6,34 m | 0 m | 6,35 m | 0 m |
| gc | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 |

The values in table 7.39 are used in the test "GNS_Type2_Message_Rec_A".

Table 7.39: Values for use in the test "GNS_Type2_Message_Rec_A"

| GNS_TYPE2_RF_IN_PARAMETERS(x) | | | | | GNS_TYPE2_APOUT_PARAMETERS(x) | | | |
|-------------------------------|----------------------------|------------------|------------------|------------------|---|--|---|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_G | add_G | add_G | add_G | add_G | add_G | add_G | add_G |
| mbi | 11111 | 10101 | 11111 | 10101 | Test GNS-B message | Normal GNS-B message | Test GNS-B message | Normal GNS-B message |
| gid | 5 7 12 12 | 12 06 06 07 | 12 05 02 12 | 05 04 04 06 | EGLL | LFPG | LEBL | EDDF |
| gmi | 1 | 1 | 1 | 1 | Type 1 message | Type 1 message | Type 1 message | Type 1 message |
| len | 27 | 27 | 27 | 27 | 27 bytes | 27 bytes | 27 bytes | 27 bytes |
| r | 1 | 0 | 2 | 0 | GNS-B installed with 3 reference receivers | GNS-B installed with 2 reference receivers | GNS-B installed with 4 reference receivers | GNS-B installed with 2 reference receivers |
| ac | 1 | 0 | 2 | 0 | GNS-B has accuracy designation B | GNS-B has accuracy designation A | GNS-B has accuracy designation C | GNS-B has accuracy designation A |
| cid | 1 | 1 | 7 | 1 | GCID 1 | GCID 1 | GNS-B not healthy | GCID 1 |
| v (bit 1) | 1 | 0 | 1 | 0 | Negative denotes west variation (counter-clockwise from true north) | Positive denotes eastward variation (clockwise from due north) | Negative denotes west variation (counter-clockwise from true north) | Positive denotes eastward variation (clockwise from due north) |
| v (bits 2 to 11) | 1 | 0 | 720 | 0 | 0,25 degrees | 0 degrees | 180,00 degrees | 0 degrees |
| vi | 1 | 0 | 255 | 0 | $0,1 \times 10^{-6}$ | 0 | $25,5 \times 10^{-6}$ | 0 |
| i (bit 8) | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| i (bits 1 to 7) | 1 | 0 | 127 | 0 | 3 | 0 | 381 | 0 |
| h | 1 | 0 | 255 | 0 | 100 | 0 | 25 500 | 0 |
| u | 1 | 0 | 255 | 0 | 1 | 0 | 255 | 0 |
| lat (bit 32) | 1 | 0 | 1 | 0 | Positive | Negative | Positive | Negative |
| lat (bits 1 to 31) | 1 | 0 | 648 000 000 | 0 | 0 degrees 0 minutes 0,0005 arcseconds | 0 degrees 0 minutes 0 arcseconds | 90 degrees 0 minutes 0 arcseconds | 0 degrees 0 minutes 0 arcseconds |
| lon (bit 32) | 1 | 0 | 1 | 0 | Positive | Negative | Positive | Negative |
| lon (bits 1 to 31) | 1 | 0 | 1 296 000 000 | 0 | 0 degrees 0 minutes 0,0005 arcseconds | 0 degrees 0 minutes 0 arcseconds | 180 degrees 0 minutes 0 arcseconds | 0 degrees 0 minutes 0 arcseconds |
| h (bit 24) | 1 | 0 | 1 | 0 | Positive | Negative | Positive | Negative |
| h (bits 1 to 23) | 1 | 0 | 8 388 607 | 0 | 0,01 m | 0 m | 83 886,07 m | 0 m |
| gc | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 |

The values in table 7.40 are used in the test "GNS_Type4_Message_Rec_A".

Table 7.40: Values for use in the test "GNS_Type4_Message_Rec_A"

| GNS_TYPE4_RF_IN_PARAMETERS(x) | | | | | GNS_TYPE4_APPOUT_PARAMETERS(x) | | | |
|----------------------------------|-------------------------|----------------|----------------|-----------------|---------------------------------------|----------------------------------|------------------------------------|----------------------------------|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| s | add_G | add_G | add_G | add_G | add_G | add_G | add_G | add_G |
| mbi | 11111 | 10101 | 11111 | 10101 | Test GNS-B message | Normal GNS-B message | Test GNS-B message | Normal GNS-B message |
| gid | 5 7 12 12 | 12 06 06 07 | 12 05 02 12 | 05 04 04 06 | EGLL | LFPG | LEBL | EDDF |
| gmi | 1 | 1 | 1 | 1 | Type 1 message | Type 1 message | Type 1 message | Type 1 message |
| len | TBD | TBD | TBD | TBD | TBD | TBD | TBD | TBD |
| l | 1 | 0 | 210 | 0 | 3 | 2 | 212 | 2 |
| Operation Type | 1 | 0 | 15 | 0 | 1 | 0 | 15 | 0 |
| SBAS provider ID | 1 | 0 | 15 | 0 | 1 | 0 | 15 | 0 |
| Airport ID | TBD | TBD | TBD | TBD | TBD | TBD | TBD | TBD |
| Runway Number | 1 | 0 | 35 | 0 | 1 | 0 | 35 | 0 |
| Runway Letter | 1 | 0 | 3 | 0 | Left runway | Single runway | Centre runway | Single runway |
| Approach performance Designator | 1 | 0 | 7 | 0 | 1 | 0 | 7 | 0 |
| Route Indicator | TBD | TBD | TBD | TBD | TBD | TBD | TBD | TBD |
| Reference Path Data Selector | 1 | 0 | 48 | 0 | 1 | 0 | 48 | 0 |
| Reference Path ID | TBD | TBD | TBD | TBD | TBD | TBD | TBD | TBD |
| LTP/FTP Latitude (bit 32) | 1 | 0 | 1 | 0 | Positive | Negative | Positive | Negative |
| LTP/FTP Latitude (bits 1 to 31) | 1 | 0 | 648 000 000 | 0 | 0 degrees 0 minutes 0,0005 arcseconds | 0 degrees 0 minutes 0 arcseconds | 90 degrees 0 minutes 0 arcseconds | 0 degrees 0 minutes 0 arcseconds |
| LTP/FTP Longitude (bit 32) | 1 | 0 | 1 | 0 | Positive | Negative | Positive | Negative |
| LTP/FTP Longitude (bits 1 to 31) | 1 | 0 | 1 296 000 000 | 0 | 0 degrees 0 minutes 0,0005 arcseconds | 0 degrees 0 minutes 0 arcseconds | 180 degrees 0 minutes 0 arcseconds | 0 degrees 0 minutes 0 arcseconds |
| LTP/FTP Height | 1 | 0 | 65 535 | 5 120 | -511,9 m | -512,0 m | 6 041,5 m | 0,0 m |
| ΔFPAP Latitude (bit 24) | 1 | 0 | 1 | 0 | Positive | Negative | Positive | Negative |
| ΔFPAP Latitude (bits 1 to 23) | 1 | 0 | 1 200 000 | 0 | 0 degrees 0 minutes 0,0005 arcseconds | 0 degrees 0 minutes 0 arcseconds | 1 degree 0 minutes 0 arcseconds | 0 degrees 0 minutes 0 arcseconds |
| ΔFPAP Longitude (bit 24) | 1 | 0 | 1 | 0 | Positive | Negative | Positive | Negative |

| GNS_TYPE4_RF_IN_PARAMETERS(x) | | | | | GNS_TYPE4_APPOUT_PARAMETERS(x) | | | |
|--|-------------------------|-----------------------|-----------------------|-----------------------|---|-------------------------------------|------------------------------------|--|
| | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) | Set P (middle range) | Set Q (min) | Set R (max) | Set S (zero) |
| ΔFPAP Longitude (bits 1 to 23) | 1 | 0 | 1 200 000 | 0 | 0 degrees 0 minutes 0,0005 arcseconds | 0 degrees 0 minutes 0 arcseconds | 1 degree 0 minutes 0 arcseconds | 0 degrees 0 minutes 0 arcseconds |
| Approach Threshold Crossing Height (TCH) | 1 | 0 | 32 707 | 0 | 0,05 | 0 | 1 635,35 | 0 |
| Approach TCH Units Selector | 1 | 1 | 1 | 1 | Metres | Metres | Metres | Metres |
| Glidepath Angle (GPA) | 1 | 0 | 9 000 | 0 | 0,01 degrees | 0 degrees | 90 degrees | 0 degrees |
| Course width | 1 | 0 | 255 | 0 | 80,25 metres | 80 metres | 143,75 metres | 80 metres |
| Δlength offset | 1 | 0 | 254 | 0 | 8 metres | 0 metres | 2 032 metres | 0 metres |
| Final Approach Segment CRC | See clause 5.5.5.2 | See clause 5.5.5.2 | See clause 5.5.5.2 | See clause 5.5.5.2 | See clause 5.5.5.2 | See clause 5.5.5.2 | See clause 5.5.5.2 | See clause 5.5.5.2 |
| va | 1 | 0 | 254 | 0 | 0,1 | 0 | 25,4 m | 0 |
| la | 1 | 0 | 254 | 0 | 0,2 | 0 | 50,8 | 0 |
| gc | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 | See clause 5.5.5 |

7.4.3.1.5 VDL4 Burst formats

A subfield value of "x" shall mean that the parameter value may be ignored for the purpose of the particular test.

The following burst formats 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 5.2.1.1.

Bursts are used in some test cases that are defined in EN 302 842-2 [5]. This clause defines additional bursts and frames not already defined in EN 302 842-2 [5].

7.4.3.1.5.1 Bursts defined for ADS-B tests.

Table 7.41: SYNC_BASIC_a (SBa): Basic variable part - Occupies one slot

| 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 | 1 | 0 |
| 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 |
| nic, cprf, b/g, tqc | 5 | nic ₄ | nic ₃ | nic ₂ | nic ₁ | cprf | b/g | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | tfom ₂ | tfom ₁ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | da ₄ | da ₃ | da ₂ | da ₁ | 0 | 0 | 0 | 0 |
| sil, lat6 | 12 | Sil ₂ | sil ₁ | lat6 ₆ | lat6 ₅ | lat6 ₄ | lat6 ₃ | lat6 ₂ | lat6 ₁ |
| br/gr, lon6 | 13 | res | br/gr | lon6 ₆ | lon6 ₅ | lon6 ₄ | lon6 ₃ | lon6 ₂ | lon6 ₁ |
| altr, bgo | 14 | altr ₉ | bgo ₇ | bgo ₆ | bgo ₅ | bgo ₄ | bgo ₃ | bgo ₂ | bgo ₁ |
| altr | 15 | altr ₈ | altr ₇ | altr ₆ | altr ₅ | altr ₄ | altr ₃ | altr ₂ | altr ₁ |
| gs | 16 | gs ₈ | gs ₇ | gs ₆ | gs ₅ | gs ₄ | gs ₃ | gs ₂ | gs ₁ |
| gs, gt | 17 | gs ₁₁ | gs ₁₀ | gs ₉ | gt ₅ | gt ₄ | gt ₃ | gt ₂ | gt ₁ |
| gt, pt | 18 | gt ₁₁ | gt ₁₀ | gt ₉ | gt ₈ | gt ₇ | gt ₆ | pt ₂ | pt ₁ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.42: SYNC_HIGH_DYNAMIC_a (SHa): High dynamic variable part - Occupies one slot

| 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 | 1 | 0 |
| 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 |
| nic, cprf, b/g, tqc | 5 | nic ₄ | nic ₃ | nic ₂ | nic ₁ | cprf | b/g | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |

| Description | Octet | Bit number | | | | | | | |
|---------------|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| balt | 8 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | tfom ₂ | tfom ₁ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | da ₄ | da ₃ | da ₂ | da ₁ | 0 | 0 | 0 | 1 |
| br/gr, bgo | 12 | br/gr | bgo ₇ | bgo ₆ | bgo ₅ | bgo ₄ | bgo ₃ | bgo ₂ | bgo ₁ |
| altr | 13 | altr ₈ | altr ₇ | altr ₆ | altr ₅ | altr ₄ | altr ₃ | altr ₂ | altr ₁ |
| altr, sil, gs | 14 | altr ₉ | sil ₃ | sil ₂ | Sil ₁ | gs ₁₂ | gs ₁₁ | gs ₁₀ | gs ₉ |
| gs | 15 | gs ₈ | gs ₇ | gs ₆ | gs ₅ | gs ₄ | gs ₃ | gs ₂ | gs ₁ |
| lon4,lat4 | 16 | lon ₄ | lon ₃ | lon ₂ | lon ₁ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| gt | 17 | gt ₈ | gt ₇ | gt ₆ | gt ₅ | gt ₄ | gt ₃ | gt ₂ | gt ₁ |
| gt,pt | 18 | gt ₁₂ | gt ₁₁ | gt ₁₀ | gt ₉ | res | res | pt ₂ | pt ₁ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.43: SYNC_FULL_POSITION_a (SPa): Full Position variable part - Occupies one slot

| 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 | 1 | 0 |
| 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 |
| nic, cprf, b/g, tqc | 5 | nic ₄ | nic ₃ | nic ₂ | nic ₁ | cprf | b/g | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | tfom ₂ | tfom ₁ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | da ₄ | da ₃ | da ₂ | da ₁ | 0 | 0 | 1 | 0 |
| pid, lat6 | 12 | pid ₁₀ | pid ₉ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| pid | 13 | pid ₈ | pid ₇ | pid ₆ | pid ₅ | pid ₄ | pid ₃ | pid ₂ | pid ₁ |
| gt, bgo | 14 | gt ₁₁ | bgo ₇ | bgo ₆ | bgo ₅ | bgo ₄ | bgo ₃ | bgo ₂ | bgo ₁ |
| gt, lon6 | 15 | gt ₁₀ | gt ₉ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| gt | 16 | gt ₈ | gt ₇ | gt ₆ | gt ₅ | gt ₄ | gt ₃ | gt ₂ | gt ₁ |
| gs | 17 | gs ₈ | gs ₇ | gs ₆ | gs ₅ | gs ₄ | gs ₃ | gs ₂ | gs ₁ |
| gs, sil, pt | 18 | gs ₁₁ | gs ₁₀ | gs ₉ | res | sil ₂ | sil ₁ | pt ₂ | pt ₁ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.44: SYNC_BASIC_GROUND_a (BGa): Basic Ground variable part - Occupies one slot

| 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 | 1 | 0 |
| 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 |
| nic, cprf, b/g, tqc | 5 | nic ₄ | nic ₃ | nic ₂ | nic ₁ | cprf | b/g | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | tfom ₂ | tfom ₁ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | da ₄ | da ₃ | da ₂ | da ₁ | 0 | 0 | 1 | 1 |
| h | 12 | res | res | res | h ₅ | h ₄ | h ₃ | h ₂ | h ₁ |
| pid, min | 13 | pid ₁₀ | pid ₉ | min ₆ | min ₅ | min ₄ | min ₃ | min ₂ | min ₁ |
| pid | 14 | pid ₈ | pid ₇ | pid ₆ | pid ₅ | pid ₄ | pid ₃ | pid ₂ | pid ₁ |
| bgo | 15 | res | bgo ₇ | bgo ₆ | bgo ₅ | bgo ₄ | bgo ₃ | bgo ₂ | bgo ₁ |
| slt | 16 | slt ₈ | slt ₇ | slt ₆ | slt ₅ | slt ₄ | slt ₃ | slt ₂ | slt ₁ |
| lon4, lat4 | 17 | lon ₄ | lon ₃ | lon ₂ | lon ₁ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| sec, pt | 18 | sec ₆ | sec ₅ | sec ₄ | sec ₃ | sec ₂ | sec ₁ | pt ₂ | pt ₁ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.45: SYNC.UTC_TIME_a (UTCa): UTC Time variable part - Occupies one slot

| 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 | 1 | 0 |
| 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 |
| nic, cprf, b/g, tqc | 5 | nic ₄ | nic ₃ | nic ₂ | nic ₁ | cprf | b/g | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | tfom ₂ | tfom ₁ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | da ₄ | da ₃ | da ₂ | da ₁ | 0 | 1 | 0 | 0 |
| day | 12 | res | res | res | day ₅ | day ₄ | day ₃ | day ₂ | day ₁ |
| yr | 13 | yr ₈ | yr ₇ | yr ₆ | yr ₅ | yr ₄ | yr ₃ | yr ₂ | yr ₁ |
| h, mon | 14 | h ₄ | h ₃ | h ₂ | h ₁ | mon ₄ | mon ₃ | mon ₂ | mon ₁ |
| h, min | 15 | res | h ₅ | min ₆ | min ₅ | min ₄ | min ₃ | min ₂ | min ₁ |
| slt | 16 | slt ₈ | slt ₇ | slt ₆ | slt ₅ | slt ₄ | slt ₃ | slt ₂ | slt ₁ |
| lon4, lat4 | 17 | lon ₄ | lon ₃ | lon ₂ | lon ₁ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| sec, pt | 18 | sec ₆ | sec ₅ | sec ₄ | sec ₃ | sec ₂ | sec ₁ | pt ₂ | pt ₁ |

| Description | Octet | Bit number | | | | | | | |
|-------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.46: SYNC_TWO_SLOT_TCP/SVQ_a (STa): Two-slot TCP/SVQ variable part - Occupies two slots

| 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 | 1 | 0 |
| 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 |
| nic, cprf, b/g, tqc | 5 | nic ₄ | nic ₃ | nic ₂ | nic ₁ | cprf | b/g | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | tfom ₂ | tfom ₁ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | da ₄ | da ₃ | da ₂ | da ₁ | 1 | 0 | 0 | 0 |
| lat | 12 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt, lat | 13 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 14 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 15 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| ttg, lon | 16 | ttg ₆ | ttg ₅ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| ttg | 17 | ttg ₄ | ttg ₃ | ttg ₂ | ttg ₁ | res | res | res | res |
| lat | 18 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt, lat | 19 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 20 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 21 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| ttg, lon | 22 | ttg ₆ | ttg ₅ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| ttg | 23 | ttg ₄ | ttg ₃ | ttg ₂ | ttg ₁ | res | res | res | res |
| lat | 24 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt, lat | 25 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 26 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 27 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| ttg, lon | 28 | ttg ₆ | ttg ₅ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| ttg | 29 | ttg ₄ | ttg ₃ | ttg ₂ | ttg ₁ | res | res | res | res |
| lat | 30 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt, lat | 31 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 32 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 33 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| ttg, lon | 34 | ttg ₆ | ttg ₅ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| ttg | 35 | ttg ₄ | ttg ₃ | ttg ₂ | ttg ₁ | res | res | res | res |
| csl | 36 | csl ₈ | csl ₇ | csl ₆ | csl ₅ | csl ₄ | csl ₃ | csl ₂ | csl ₁ |
| csl | 37 | csl ₁₆ | csl ₁₅ | csl ₁₄ | csl ₁₃ | csl ₁₂ | csl ₁₁ | csl ₁₀ | csl ₉ |
| st, csl | 38 | st ₃ | st ₂ | st ₁ | csl ₂₁ | csl ₂₀ | csl ₁₉ | csl ₁₈ | csl ₁₇ |
| csr | 39 | csr ₈ | csr ₇ | csr ₆ | csr ₅ | csr ₄ | csr ₃ | csr ₂ | csr ₁ |
| csr | 40 | csr ₁₆ | csr ₁₅ | csr ₁₄ | csr ₁₃ | csr ₁₂ | csr ₁₁ | csr ₁₀ | csr ₉ |
| nacv, csr | 41 | nacv ₃ | nacv ₂ | nacv ₁ | csr ₂₁ | csr ₂₀ | csr ₁₉ | csr ₁₈ | csr ₁₇ |

| Description | Octet | Bit number | | | | | | | |
|----------------------------|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| nacp, cdti, nicb, acas, ra | 42 | nacp ₄ | nacp ₃ | nacp ₂ | nacp ₁ | cdti | nicb | acas | ra |
| pid | 43 | pid ₈ | pid ₇ | pid ₆ | pid ₅ | pid ₄ | pid ₃ | pid ₂ | pid ₁ |
| pid, lat6 | 44 | pid ₁₀ | pid ₉ | lat6 ₆ | lat6 ₅ | lat6 ₄ | lat6 ₃ | lat6 ₂ | lat6 ₁ |
| altr, nucl, br/gr, lon6 | 45 | altr ₉ | br/gr | lon6 ₆ | lon6 ₅ | lon6 ₄ | lon6 ₃ | lon6 ₂ | lon6 ₁ |
| altr | 46 | altr ₈ | altr ₇ | altr ₆ | altr ₅ | altr ₄ | altr ₃ | altr ₂ | altr ₁ |
| sil, ac | 47 | sil ₃ | sil ₂ | sil ₁ | ac ₅ | ac ₄ | ac ₃ | ac ₂ | ac ₁ |
| gs | 48 | gs ₈ | gs ₇ | gs ₆ | gs ₅ | gs ₄ | gs ₃ | gs ₂ | gs ₁ |
| gs, gt | 49 | gs ₁₁ | gs ₁₀ | gs ₉ | gt ₅ | gt ₄ | gt ₃ | gt ₂ | gt ₁ |
| gt, pt | 50 | gt ₁₁ | gt ₁₀ | gt ₉ | gt ₈ | gt ₇ | gt ₆ | pt ₂ | pt ₁ |
| po | 51 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 52 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 53 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.47: SYNC_SINGLE_SLOT_TCP_a (SSa): Single Slot TCP variable part - Occupies one slot

| 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 | 1 | 0 |
| 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 |
| nic, cprf, b/g, tqc | 5 | nic ₄ | nic ₃ | nic ₂ | nic ₁ | cprf | b/g | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | tfom ₂ | tfom ₁ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | da ₄ | da ₃ | da ₂ | da ₁ | 1 | 0 | 0 | 1 |
| balt | 12 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| balt, no, pid | 13 | balt ₁₂ | balt ₁₁ | balt ₁₀ | no ₂ | no ₁ | pid ₃ | pid ₂ | pid ₁ |
| pid | 14 | pid ₈ | pid ₇ | pid ₆ | pid ₅ | pid ₄ | pid ₃ | pid ₂ | pid ₁ |
| lat | 15 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| lat, lon | 16 | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ |
| lon | 17 | lon ₁₀ | lon ₉ | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ |
| ttg, pt | 18 | ttg ₆ | ttg ₅ | ttg ₄ | ttg ₃ | ttg ₂ | ttg ₁ | pt ₂ | pt ₁ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.48: SYNC_SINGLE_SLOT_SVQ_a (SVa): Single Slot SVQ variable part - Occupies one slot

| 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 | 1 | 0 |
| 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 |
| nic, cprf, b/g, tqc | 5 | nic ₄ | nic ₃ | nic ₂ | nic ₁ | cprf | b/g | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | tfom ₂ | tfom ₁ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | da ₄ | da ₃ | da ₂ | da ₁ | 0 | 1 | 0 | 1 |
| nacp | 12 | 0 | 0 | 0 | 1 | nacp ₄ | nacp ₃ | nacp ₂ | nacp ₁ |
| nacv, sil, nicb, acas, ra | 13 | nacv ₃ | nacv ₂ | nacv ₁ | sil ₂ | sil ₁ | nicb | acas | ra |
| st, ac | 14 | st ₃ | st ₂ | st ₁ | ac ₅ | ac ₄ | ac ₃ | ac ₂ | ac ₁ |
| res | 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| res | 16 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| res | 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| res, pt | 18 | 0 | 0 | 0 | 0 | 0 | 0 | pt ₂ | pt ₁ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.49: SYNC_AIRCRAFT_DATA_a (SDa): Aircraft data variable part - Occupies one slot

| 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 | 1 | 0 |
| 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 |
| nic, cprf, b/g, tqc | 5 | nic ₄ | nic ₃ | nic ₂ | nic ₁ | cprf | b/g | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | tfom ₂ | tfom ₁ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | da ₄ | da ₃ | da ₂ | da ₁ | 1 | 0 | 1 | 0 |
| ac | 12 | 0 | 0 | 0 | 1 | ac ₄ | ac ₃ | ac ₂ | ac ₁ |
| ac, st, csl | 13 | ac ₅ | st ₃ | st ₂ | st ₁ | csl ₁₂ | csl ₁₁ | csl ₁₀ | csl ₉ |
| csl | 14 | csl ₈ | csl ₇ | csl ₆ | csl ₅ | csl ₄ | csl ₃ | csl ₂ | csl ₁ |
| csl | 15 | csl ₂₀ | csl ₁₉ | csl ₁₈ | csl ₁₇ | csl ₁₆ | csl ₁₅ | csl ₁₄ | csl ₁₃ |
| csl, csr | 16 | csl ₂₁ | csr ₇ | csr ₆ | csr ₅ | csr ₄ | csr ₃ | csr ₂ | csr ₁ |
| csr | 17 | csr ₁₅ | csr ₁₄ | csr ₁₃ | csr ₁₂ | csr ₁₁ | csr ₁₀ | csr ₉ | csr ₈ |
| csr, pt | 18 | csr ₂₁ | csr ₂₀ | csr ₁₉ | csr ₁₈ | csr ₁₇ | csr ₁₆ | pt ₂ | pt ₁ |

| Description | Octet | Bit number | | | | | | | |
|-------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.50: SYNC_HIGH_RESOLUTION_a (SRa): High resolution variable part - Occupies one slot

| 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 | 1 | 0 |
| 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 |
| nic, cprf, b/g, tqc | 5 | nic ₄ | nic ₃ | nic ₂ | nic ₁ | cprf | b/g | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | balt ₁₂ | balt ₁₁ | balt ₁₀ | balt ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | balt ₈ | balt ₇ | balt ₆ | balt ₅ | balt ₄ | balt ₃ | balt ₂ | balt ₁ |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | tfom ₂ | tfom ₁ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | da ₄ | da ₃ | da ₂ | da ₁ | 1 | 0 | 1 | 0 |
| id | 12 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| sil, gs | 13 | res | res | sil ₂ | sil ₁ | gs ₁₂ | gs ₁₁ | gs ₁₀ | gs ₉ |
| gs | 14 | gs ₈ | gs ₇ | gs ₆ | gs ₅ | gs ₄ | gs ₃ | gs ₂ | gs ₁ |
| lon8 | 15 | lon8 ₈ | lon8 ₇ | lon8 ₆ | lon8 ₅ | lon8 ₄ | lon8 ₃ | lon8 ₂ | lon8 ₁ |
| lat8 | 16 | lat8 ₈ | lat8 ₇ | lat8 ₆ | lat8 ₅ | lat8 ₄ | lat8 ₃ | lat8 ₂ | lat8 ₁ |
| gt | 17 | gt ₈ | gt ₇ | gt ₆ | gt ₅ | gt ₄ | gt ₃ | gt ₂ | gt ₁ |
| gt, tind, pt | 18 | gt ₁₂ | gt ₁₁ | gt ₁₀ | gt ₉ | tind ₂ | tind ₁ | pt ₂ | pt ₁ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.51: DIR_REQ_a (Da): Contains general request

| 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 |
| r-mi, mi | 5 | r-mi ₁ | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| res, r-mi | 6 | 0 | 0 | r-mi ₇ | r-mi ₆ | r-mi ₅ | r-mi ₄ | r-mi ₃ | r-mi ₂ |
| dt, f | 7 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 8 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| lg | 9 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| lg, res, do | 10 | res | res | trmt | do ₁₃ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 11 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| or, rcvr, pr_flag, nr | 12 | or | rcvr ₂ | rcvr ₁ | pr_flag | nr ₄ | nr ₃ | nr ₂ | nr 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 | 1 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 17 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 18 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.52: DIR_SYNC_BURST_a (DSa): Directed sync burst - Occupies one slot. Basic variable part

| 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 | 1 | 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 |
| nucp, cprf, b/g, tqc | 5 | x | x | x | x | x | x | 1 | 0 |
| lat | 6 | x | x | x | x | x | x | x | x |
| balt | 7 | x | x | x | x | x | x | x | x |
| balt | 8 | x | x | x | x | x | x | x | x |
| lon | 9 | x | x | x | x | x | x | x | x |
| tfom, lon | 10 | x | x | x | x | x | x | x | x |
| da, id | 11 | x | x | x | x | 0 | 0 | 0 | 0 |
| in | 12 | x | x | x | x | x | x | x | x |
| in | 13 | x | x | x | x | x | x | x | x |
| in | 14 | x | x | x | x | x | x | x | x |
| in | 15 | x | x | x | x | x | x | x | x |
| in | 16 | x | x | x | x | x | x | x | x |
| in | 17 | x | x | x | x | x | x | x | x |
| in, pt | 18 | x | x | x | x | x | x | pt ₂ | pt ₁ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.53: PLEA_a (Pa): Information field contains destination address - Fits within delayed burst

| 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 |
| in, mi | 5 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| d | 6 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 7 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 8 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| erid, d | 9 | 0 | 0 | 0 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 10 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 11 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.54: PLEA_RESP_a (PRa): Directed request with pr_flag = 1, nr ≠ "special"

| 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 |
| res, mi | 5 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| a | 6 | a _{11,6} | a _{11,5} | a _{8,6} | a _{8,5} | a _{8,4} | a _{8,3} | a _{8,2} | a _{8,1} |
| a | 7 | a _{11,4} | a _{11,3} | a _{7,6} | a _{7,5} | a _{7,4} | a _{7,3} | a _{7,2} | a _{7,1} |
| a | 8 | a _{11,2} | a _{11,1} | a _{6,6} | a _{6,5} | a _{6,4} | a _{6,3} | a _{6,2} | a _{6,1} |
| a | 9 | a _{10,6} | a _{10,5} | a _{5,6} | a _{5,5} | a _{5,4} | a _{5,3} | a _{5,2} | a _{5,1} |
| a | 10 | a _{10,4} | a _{10,3} | a _{4,6} | a _{4,5} | a _{4,4} | a _{4,3} | a _{4,2} | a _{4,1} |
| a | 11 | a _{10,2} | a _{10,1} | a _{3,6} | a _{3,5} | a _{3,4} | a _{3,3} | a _{3,2} | a _{3,1} |
| a | 12 | a _{9,6} | a _{9,5} | a _{2,6} | a _{2,5} | a _{2,4} | a _{2,3} | a _{2,2} | a _{2,1} |
| a | 13 | a _{9,4} | a _{9,3} | a _{1,6} | a _{1,5} | a _{1,4} | a _{1,3} | a _{1,2} | a _{1,1} |
| a, off | 14 | a _{9,2} | a _{9,1} | off ₉ | off ₈ | off ₇ | off ₆ | off ₅ | off ₄ |
| off, pr_flag, nr | 15 | off ₃ | off ₂ | off ₁ | 1 | nr ₄ | nr ₃ | nr ₂ | nr ₁ |
| 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 | 1 | 1 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.55: ADSB_REQUEST_a (ARa): Includes unicast reservation, sleep = 0, auto = 0

| 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 |
| sleep, auto, r-b/a, burst ID | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| d | 6 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| d | 7 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ |
| d | 8 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ |
| sdf, d | 9 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ |
| ro | 10 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ |
| lg | 11 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| pr | 12 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 13 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 14 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.56: ADSB_REQUEST_b (ARb): Includes unicast reservation, sleep = 0, auto = 1

| 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 |
| sleep, auto, r-b/a, burst ID | 5 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg, aux, r-id | 6 | lg ₂ | lg ₁ | aux ₂ | aux ₁ | r-id ₄ | r-id ₃ | r-id ₂ | r-id 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, d | 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 |
| pr | 13 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ |
| c | 14 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 15 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.57: ADSB_REQUEST_c (ARc): Includes directed request reservation, sleep = 1, auto = 0

| 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 |
| sleep, auto, r-b/a, burst ID | 5 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| snr, vel | 6 | snr ₄ | snr ₃ | snr ₂ | snr ₁ | vel ₄ | vel ₃ | vel ₂ | vel ₁ |
| pos | 7 | pos ₈ | pos ₇ | pos ₆ | pos ₅ | pos ₄ | pos ₃ | pos ₂ | pos ₁ |
| dt | 8 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 9 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| lg | 10 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| trmt | 11 | res | res | trmt | do ₁₃ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 12 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| or; rcvr, nr; pr_flag = 0 | 13 | or | rcvr ₂ | rcvr ₁ | 0 | nr ₄ | nr ₃ | nr ₂ | nr 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 | 17 | 0 | 1 | 1 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 18 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 19 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.58: ADSB_REQUEST_d (ARd) Includes directed request reservation, sleep = 1, auto = 1

| 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 |
| sleep, auto, r-b/a, burst ID | 5 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| snr, vel | 6 | snr ₄ | snr ₃ | snr ₂ | snr ₁ | vel ₄ | vel ₃ | vel ₂ | vel ₁ |
| pos | 7 | pos ₈ | pos ₇ | pos ₆ | pos ₅ | pos ₄ | pos ₃ | pos ₂ | pos ₁ |
| lg, aux, r-id | 8 | lg ₂ | lg ₁ | aux ₂ | aux ₁ | r-id ₄ | r-id ₃ | r-id ₂ | r-id ₁ |
| dt | 9 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 10 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| lg | 11 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ |
| trmt | 12 | res | res | trmt | do ₁₃ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 13 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| or; rcvr, nr; pr_flag = 0 | 14 | or | rcvr ₂ | rcvr ₁ | 0 | nr ₄ | nr ₃ | nr ₂ | nr ₁ |
| d | 15 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ |
| 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 |
| erid | 18 | 0 | 1 | 1 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ |
| c | 19 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 20 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.59: ADSB_REQUEST_e (ARE) Includes directed request reservation, sleep = 0, auto = 1

| Description | Octet | Bit number | | | | | | | | |
|------------------------------|-------|-----------------|-------------------|-------------------|------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| s, ver, rid, a/d | 1 | s ₂₇ | s ₂₆ | s ₂₅ | 0 | 0 | 0 | 0 | 0 | 1 |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ | s ₁₆ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ | s ₈ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ | s ₀ |
| sleep, auto, r-b/a, burst ID | 5 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| lg, aux, r-id | 6 | lg ₂ | lg ₁ | aux ₂ | aux ₁ | r-id ₄ | r-id ₃ | r-id ₂ | r-id ₁ | r-id ₀ |
| dt | 7 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ | f ₈ |
| f | 8 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ | f ₀ |
| lg | 9 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | lg ₀ |
| trmt | 10 | res | res | trmt | do ₁₃ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ | do ₈ |
| do | 11 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ | do ₀ |
| or; rcvr, nr; pr_flag = 0 | 12 | or | rcvr ₂ | rcvr ₁ | 0 | nr ₄ | nr ₃ | nr ₂ | nr ₁ | nr ₀ |
| d | 13 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ | d ₁₆ |
| d | 14 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ | d ₈ |
| d | 15 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ | d ₀ |
| erid | 16 | 0 | 1 | 1 | 0 | 0 | d ₂₇ | d ₂₆ | d ₂₅ | d ₂₄ |
| c | 17 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ | c ₈ |
| c | 18 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ | c ₀ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.60: DIR_SYNC_BURST_b (DSb): Directed sync burst - Occupies one slot - Full position variable part

| Description | Octet | Bit number | | | | | | | | |
|----------------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| s, ver, rid, a/d | 1 | s ₂₇ | s ₂₆ | s ₂₅ | 0 | 0 | 0 | 1 | 1 | 1 |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ | s ₁₆ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ | s ₈ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ | s ₀ |
| nucp, cprf, b/g, tqc | 5 | x | x | x | x | x | x | 1 | 0 | 0 |
| lat | 6 | x | x | x | x | x | x | x | x | x |
| balt | 7 | x | x | x | x | x | x | x | x | x |
| balt | 8 | x | x | x | x | x | x | x | x | x |
| lon | 9 | x | x | x | x | x | x | x | x | x |
| tfom, lon | 10 | x | x | x | x | x | x | x | x | x |
| da, id | 11 | x | x | x | x | 0 | 0 | 1 | 0 | 0 |
| in | 12 | x | x | x | x | x | x | x | x | x |
| in | 13 | x | x | x | x | x | x | x | x | x |
| in | 14 | x | x | x | x | x | x | x | x | x |
| in | 15 | x | x | x | x | x | x | x | x | x |
| in | 16 | x | x | x | x | x | x | x | x | x |
| in | 17 | x | x | x | x | x | x | x | x | x |
| in, pt | 18 | x | x | x | x | x | x | pt ₂ | pt ₁ | pt ₀ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ | po ₀ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ | c ₈ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ | c ₀ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.61: DIR_SYNC_BURST_c (DSc): Directed sync burst - Occupies one slot - Two-slot TCP/SVQ variable part

| 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 | 1 | 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 |
| nucp, cprf, b/g, tqc | 5 | x | x | x | x | x | x | 1 | 0 |
| lat | 6 | x | x | x | x | x | x | x | x |
| balt | 7 | x | x | x | x | x | x | x | x |
| balt | 8 | x | x | x | x | x | x | x | x |
| lon | 9 | x | x | x | x | x | x | x | x |
| tfom, lon | 10 | x | x | x | x | x | x | x | x |
| da, id | 11 | x | x | x | x | 1 | 0 | 0 | 0 |
| in | 12 | x | x | x | x | x | x | x | x |
| in | 13 | x | x | x | x | x | x | x | x |
| in | 14 | x | x | x | x | x | x | x | x |
| in | 15 | x | x | x | x | x | x | x | x |
| in | 16 | x | x | x | x | x | x | x | x |
| in | 17 | x | x | x | x | x | x | x | x |
| in | 18 | x | x | x | x | x | x | x | x |
| in | 19 | x | x | x | x | x | x | x | x |
| in | 20 | x | x | x | x | x | x | x | x |
| in | 21 | x | x | x | x | x | x | x | x |
| in | 22 | x | x | x | x | x | x | x | x |
| in | 23 | x | x | x | x | x | x | x | x |
| in | 24 | x | x | x | x | x | x | x | x |
| in | 25 | x | x | x | x | x | x | x | x |
| in | 26 | x | x | x | x | x | x | x | x |
| in | 27 | x | x | x | x | x | x | x | x |
| in | 28 | x | x | x | x | x | x | x | x |
| in | 29 | x | x | x | x | x | x | x | x |
| in | 30 | x | x | x | x | x | x | x | x |
| in | 31 | x | x | x | x | x | x | x | x |
| in | 32 | x | x | x | x | x | x | x | x |
| in | 33 | x | x | x | x | x | x | x | x |
| in | 34 | x | x | x | x | x | x | x | x |
| in | 35 | x | x | x | x | x | x | x | x |
| in | 36 | x | x | x | x | x | x | x | x |
| in | 37 | x | x | x | x | x | x | x | x |
| in | 38 | x | x | x | x | x | x | x | x |
| in | 39 | x | x | x | x | x | x | x | x |
| in | 40 | x | x | x | x | x | x | x | x |
| in | 41 | x | x | x | x | x | x | x | x |
| in | 42 | x | x | x | x | x | x | x | x |
| in | 43 | x | x | x | x | x | x | x | x |
| in | 44 | x | x | x | x | x | x | x | x |
| in | 45 | x | x | x | x | x | x | x | x |
| in | 46 | x | x | x | x | x | x | x | x |
| in | 47 | x | x | x | x | x | x | x | x |
| in | 48 | x | x | x | x | x | x | x | x |
| in | 49 | x | x | x | x | x | x | x | x |
| in, pt | 50 | x | x | x | x | x | x | pt ₂ | pt ₁ |
| po | 51 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 52 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 53 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.62: CTRL_CHAN_a (Ca): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, no transmission block, 1 frequency block, no parameter block and 1 script block) - Fits within one slot

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| sleep, auto, rcv, f | 13 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 14 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| frq, txd | 15 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 16 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 17 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 18 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.63: CTRL_CHAN_b (Cb): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 destination block, no transmission block, 1 frequency block, no parameter block and 1 script block)

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| ido, d | 13 | 0 | ido ₄ | ido ₃ | ido ₂ | ido ₁ | d ₂₇ | d ₂₆ | d ₂₅ |
| 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 |
| sleep, auto, rcv, f | 17 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 18 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| frq, txd | 19 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 20 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |

| Description | Octet | Bit number | | | | | | | |
|-------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| c | 21 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 22 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.64: CTRL_CHAN_c (Cc): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 destination block, no transmission block, 1 frequency block, no parameter block and 2 script blocks)

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| ido, d | 13 | 0 | ido ₄ | ido ₃ | ido ₂ | ido ₁ | d ₂₇ | d ₂₆ | d ₂₅ |
| 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 |
| sleep, auto, rcv, f | 17 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 18 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| frq(1), txd(1) | 19 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| frq(2), txd(2) | 20 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 21 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 22 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 23 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.65: CTRL_CHAN_d (Cd): Channel management parameter in UCTRL burst with response reservation (1 header block, 2 destination blocks, no transmission block, 1 frequency block, no parameter block and 1 script block)

| Description | Octet | Bit number | | | | | | | | |
|---------------------|-------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|------------------|------------------|---|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| s, ver, rid, a/d | 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 | |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ | |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ | |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ | |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ | |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ | |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ | |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ | |
| ido, d | 13 | 0 | ido ₄ | ido ₃ | ido ₂ | ido ₁ | d ₂₇ | d ₂₆ | d ₂₅ | |
| 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 | |
| ido, d | 17 | 0 | ido ₄ | ido ₃ | ido ₂ | ido ₁ | d ₂₇ | d ₂₆ | d ₂₅ | |
| d | 18 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ | |
| d | 19 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ | |
| d | 20 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ | |
| sleep, auto, rcv, f | 21 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ | |
| f | 22 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ | |
| frq, txd | 23 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ | |
| erid, d | 24 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | |
| c | 25 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ | |
| c | 26 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ | |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.66: CTRL_CHAN_e (Ce): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 multicast destination block, no transmission block, 1 frequency block, no parameter block and 1 script block)

| Description | Octet | Bit number | | | | | | | | |
|--------------------|-------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|------------------|-----------------|---|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| s, ver, rid, a/d | 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 | |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ | |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ | |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ | |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ | |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ | |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ | |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ | |

| Description | Octet | Bit number | | | | | | | |
|-------------------------|-------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| vc | 13 | 0 | res | lon ₁₄ | lon ₁₃ | vc ₄ | vc ₃ | vc ₂ | vc ₁ |
| lat, lat4 | 14 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| | 15 | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| lon, lon4 | 16 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| | 17 | lon ₄ ₄ | lon ₄ ₃ | lon ₄ ₂ | lon ₄ ₁ | lat ₄ ₄ | lat ₄ ₃ | lat ₄ ₂ | lat ₄ ₁ |
| lalt | 18 | lalt ₈ | lalt ₇ | lalt ₆ | lalt ₅ | lalt ₄ | lalt ₃ | lalt ₂ | lalt ₁ |
| ualt | 19 | ualt ₈ | ualt ₇ | ualt ₆ | ualt ₅ | ualt ₄ | ualt ₃ | ualt ₂ | ualt ₁ |
| vertex 1: radial 1 (r1) | 20 | r _{1,8} | r _{1,7} | r _{1,6} | r _{1,5} | r _{1,4} | r _{1,3} | r _{1,2} | r _{1,1} |
| distance 1 (d1) | 21 | d _{1,8} | d _{1,7} | d _{1,6} | d _{1,5} | d _{1,4} | d _{1,3} | d _{1,2} | d _{1,1} |
| vertex 2: radial 2 (r2) | 22 | r _{2,8} | r _{2,7} | r _{2,6} | r _{2,5} | r _{2,4} | r _{2,3} | r _{2,2} | r _{2,1} |
| distance 2 (d2) | 23 | d _{2,8} | d _{2,7} | d _{2,6} | d _{2,5} | d _{2,4} | d _{2,3} | d _{2,2} | d _{2,1} |
| sleep, auto, rcv, f | 24 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 25 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| frq, txd | 26 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 27 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 28 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 29 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.67: CTRL_CHAN_f (Cf): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 destination block, no transmission block, 2 frequency blocks, no parameter block and 2 script blocks)

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| ido, d | 13 | 0 | ido ₄ | ido ₃ | ido ₂ | ido ₁ | d ₂₇ | d ₂₆ | d ₂₅ |
| 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 |
| sleep(1), auto(1), rcv(1), f(1) | 17 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f(1) | 18 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| sleep(2), auto(2), rcv(2), f(2) | | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f(2) | | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |

| Description | Octet | Bit number | | | | | | | |
|----------------|-------|------------------|------------------|------------------|-----------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| frq(1), txd(1) | 19 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| frq(2), txd(2) | 20 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 21 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 22 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 23 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.68: CTRL_CHAN_g (Cg): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, 1 transmission block, 1 frequency block, no parameter block and 2 script blocks) - Fits within one slot

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| id, aux, b/g | 13 | id ₄ | id ₃ | id ₂ | id ₁ | aux | b/g | res | 0 |
| sleep, auto, rcv, f | 14 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 15 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| frq(1), txd(1) | 16 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| frq(2), txd(2) | 17 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 18 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 19 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 20 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.69: CTRL_CHAN_h (Ch): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, 2 transmission blocks, 1 frequency block, no parameter block and 2 script blocks) - Fits within one slot

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| id(1), aux(1), b/g(1) | 13 | id ₄ | id ₃ | id ₂ | id ₁ | aux | b/g | res | 0 |
| id(2), aux(2), b/g(2) | 14 | id ₄ | id ₃ | id ₂ | id ₁ | aux | b/g | res | 0 |
| sleep, auto, rcv, f | 15 | sleep | auto | rcv | res | 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 |
| frq(1), txd(1) | 17 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| frq(2), txd(2) | 18 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 19 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.70: CTRL_CHAN_i (Ci): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, 2 transmission blocks (1 non-sync burst), 1 frequency block, no parameter block and 2 script blocks)

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| id(1), aux(1), b/g(1) | 13 | id ₄ | id ₃ | id ₂ | id ₁ | aux | b/g | res | 0 |
| mid | 14 | mid ₈ | mid ₇ | mid ₆ | mid ₅ | mid ₄ | mid ₃ | mid ₂ | mid ₁ |
| app, len | 15 | app | app | app | app | len ₄ | len ₃ | len ₂ | len ₁ |
| sleep, auto, rcv, f | 16 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 17 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |

| Description | Octet | Bit number | | | | | | | |
|----------------|-------|------------------|------------------|------------------|-----------------|------------------|------------------|------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| frq(1), txd(1) | 18 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| frq(2), txd(2) | 19 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 20 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 21 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 22 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.71: CTRL_CHAN_j (Cj): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 destination block, 1 transmission block, 1 frequency block, no parameter block and 1 script block)

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| ido, d | 13 | 0 | ido ₄ | ido ₃ | ido ₂ | ido ₁ | d ₂₇ | d ₂₆ | d ₂₅ |
| 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 |
| id, aux, b/g | 17 | id ₄ | id ₃ | id ₂ | id ₁ | aux | b/g | res | 0 |
| sleep, auto, rcv, f | 18 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 19 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| frq, txd | 20 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 21 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 22 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 23 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.72: CTRL_CHAN_k (Ck): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, 1 transmission block, 1 frequency block, no parameter block and 1 script blocks (with 1 repeat)) - Fits within one slot

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| id, aux, b/g | 13 | id ₄ | id ₃ | id ₂ | id ₁ | aux | b/g | res | 0 |
| sleep, auto, rcv, f | 14 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 15 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| rpt | 16 | 1 | 1 | 1 | rpt ₅ | rpt ₄ | rpt ₃ | rpt ₂ | rpt ₁ |
| frq, txd | 17 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 18 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 19 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 20 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.73: CTRL_CHAN_I (CI): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, 1 transmission block, 1 frequency block, no parameter block and 1 script blocks (with 2 invalid repeats)). Fits within one slot

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| id, aux, b/g | 13 | id ₄ | id ₃ | id ₂ | id ₁ | aux | b/g | res | 0 |
| sleep, auto, rcv, f | 14 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 15 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| rpt | 16 | 1 | 1 | 1 | rpt ₅ | rpt ₄ | rpt ₃ | rpt ₂ | rpt ₁ |
| rpt | 17 | 1 | 1 | 1 | rpt ₅ | rpt ₄ | rpt ₃ | rpt ₂ | rpt ₁ |
| frq, txd | 18 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |

| Description | Octet | Bit number | | | | | | | |
|-------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| erid, d | 19 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.74: CTRL_CHAN_m (Cm): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 destination block, no transmission block, 1 frequency block, 1 parameter block and 1 script block)

| 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 | 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 |
| UCTRL DLPDU header | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| dt | 6 | dt ₄ | dt ₃ | dt ₂ | dt ₁ | do ₁₂ | do ₁₁ | do ₁₀ | do ₉ |
| do | 7 | do ₈ | do ₇ | do ₆ | do ₅ | do ₄ | do ₃ | do ₂ | do ₁ |
| dc | 8 | dc ₈ | dc ₇ | dc ₆ | dc ₅ | dc ₄ | dc ₃ | dc ₂ | dc ₁ |
| csid, tc | 9 | csid ₄ | csid ₃ | csid ₂ | csid ₁ | tc ₄ | tc ₃ | tc ₂ | tc ₁ |
| fc, styp, sr | 10 | fc ₃ | fc ₂ | fc ₁ | styp | sr ₄ | sr ₃ | sr ₂ | sr ₁ |
| pc | 11 | res | res | pc ₆ | pc ₅ | pc ₄ | pc ₃ | pc ₂ | pc ₁ |
| sc | 12 | sc ₈ | sc ₇ | sc ₆ | sc ₅ | sc ₄ | sc ₃ | sc ₂ | sc ₁ |
| ido, d | 13 | 0 | ido ₄ | ido ₃ | ido ₂ | ido ₁ | d ₂₇ | d ₂₆ | d ₂₅ |
| 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 |
| sleep, auto, rcv, f | 17 | sleep | auto | rcv | res | f ₁₂ | f ₁₁ | f ₁₀ | f ₉ |
| f | 18 | f ₈ | f ₇ | f ₆ | f ₅ | f ₄ | f ₃ | f ₂ | f ₁ |
| Q2a | 19 | Q2a ₈ | Q2a ₇ | Q2a ₆ | Q2a ₅ | Q2a ₄ | Q2a ₃ | Q2a ₂ | Q2a ₁ |
| Q2b | 20 | Q2b ₈ | Q2b ₇ | Q2b ₆ | Q2b ₅ | Q2b ₄ | Q2b ₃ | Q2b ₂ | Q2b ₁ |
| Q2c | 21 | Q2c ₈ | Q2c ₇ | Q2c ₆ | Q2c ₅ | Q2c ₄ | Q2c ₃ | Q2c ₂ | Q2c ₁ |
| Q2d | 22 | Q2d ₈ | Q2d ₇ | Q2d ₆ | Q2d ₅ | Q2d ₄ | Q2d ₃ | Q2d ₂ | Q2d ₁ |
| Q2a, Q2b, Q2c, Q2d | 23 | Q2d ₁₀ | Q2d ₉ | Q2c ₁₀ | Q2c ₉ | Q2b ₁₀ | Q2b ₉ | Q2a ₁₀ | Q2a ₉ |
| Q1, Q4 | 24 | Q1 ₃ | Q1 ₂ | Q1 ₁ | Q4 ₅ | Q4 ₄ | Q4 ₃ | Q4 ₂ | Q4 ₁ |
| Q1, V12 | 25 | Q1 ₄ | V12 ₇ | V12 ₆ | V12 ₅ | V12 ₄ | V12 ₃ | V12 ₂ | V12 ₁ |
| TV11min, TV11max | 26 | TV11m in ₄ | TV11m in ₃ | TV11m in ₂ | TV11m in ₁ | TV11m ax ₄ | TV11m ax ₃ | TV11m ax ₂ | TV11m ax ₁ |
| V11 | 27 | res | res | V11 ₆ | V11 ₅ | V11 ₄ | V11 ₃ | V11 ₂ | V11 ₁ |
| VS2 | 28 | res | res | VS2 ₆ | VS2 ₅ | VS2 ₄ | VS2 ₃ | VS2 ₂ | VS2 ₁ |
| VS4 | 29 | res | VS4 ₇ | VS4 ₆ | VS4 ₅ | VS4 ₄ | VS4 ₃ | VS4 ₂ | VS4 ₁ |
| frq, txd | 30 | frq ₃ | frq ₂ | frq ₁ | nsd | txd ₄ | txd ₃ | txd ₂ | txd ₁ |
| erid, d | 31 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 32 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 33 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.75: SYNC_BURST_g (Sg): Occupies one slot. Autonomous burst - Basic variable part - Lat6 and lon6 specified. "x" = do not care

| 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 | 1 | 0 |
| 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 |
| nucp, cprf, b/g, tqc | 5 | x | x | x | x | cprf | x | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | x | x | x | x | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | x | x | x | x | x | x | x | x |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | x | x | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | x | x | x | x | 0 | 0 | 0 | 0 |
| in | 12 | x | x | lat ₆ ₆ | lat ₆ ₅ | lat ₆ ₄ | lat ₆ ₃ | lat ₆ ₂ | lat ₆ ₁ |
| in | 13 | x | x | lon ₆ ₆ | lon ₆ ₅ | lon ₆ ₄ | lon ₆ ₃ | lon ₆ ₂ | lon ₆ ₁ |
| in | 14 | x | x | x | x | x | x | x | x |
| in | 15 | x | x | x | x | x | x | x | x |
| in | 16 | x | x | x | x | x | x | x | x |
| in | 17 | x | x | x | x | x | x | x | x |
| in, pt | 18 | x | x | x | x | x | x | pt ₂ | pt ₁ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.76: SYNC_BURST_h (Sh): Occupies one slot - Autonomous burst, basic ground variable part Lat4, lon4, and pid specified. "x" = do not care

| 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 | 1 | 0 |
| 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 |
| nucp, cprf, b/g, tqc | 5 | x | x | x | x | cprf | x | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | x | x | x | x | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | x | x | x | x | x | x | x | x |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | x | x | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | x | x | x | x | 0 | 0 | 1 | 1 |
| in | 12 | x | x | x | x | x | x | x | x |
| in | 13 | pid ₁₀ | pid ₉ | x | x | x | x | x | x |
| in | 14 | pid ₈ | pid ₇ | pid ₆ | pid ₅ | pid ₄ | pid ₃ | pid ₂ | pid ₁ |
| in | 15 | x | x | x | x | x | x | x | x |
| in | 16 | x | x | x | x | x | x | x | x |
| in | 17 | lon ₄ ₄ | lon ₄ ₃ | lon ₄ ₂ | lon ₄ ₁ | lat ₄ ₄ | lat ₄ ₃ | lat ₄ ₂ | lat ₄ ₁ |
| in, pt | 18 | x | x | x | x | x | x | pt ₂ | pt ₁ |

| Description | Octet | Bit number | | | | | | | |
|-------------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.77: SYNC_BURST_i (Si): Occupies one slot - Autonomous burst, high resolution variable part - Lat8 and lon8 specified. "x" = do not care

| 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 | 1 | 0 |
| 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 |
| nucp, cprf, b/g, tqc | 5 | x | x | x | x | cprf | x | 1 | 0 |
| lat | 6 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| balt | 7 | x | x | x | x | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| balt | 8 | x | x | x | x | x | x | x | x |
| lon | 9 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| tfom, lon | 10 | x | x | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| da, id | 11 | x | x | x | x | 1 | 0 | 1 | 0 |
| in | 12 | x | x | x | x | x | x | x | x |
| in | 13 | x | x | x | x | x | x | x | x |
| in | 14 | x | x | x | x | x | x | x | x |
| in | 15 | lon8 ₈ | lon8 ₇ | lon8 ₆ | lon8 ₅ | lon8 ₄ | lon8 ₃ | lon8 ₂ | lon8 ₁ |
| in | 16 | lat8 ₈ | lat8 ₇ | lat8 ₆ | lat8 ₅ | lat8 ₄ | lat8 ₃ | lat8 ₂ | lat8 ₁ |
| in | 17 | x | x | x | x | x | x | x | x |
| in, pt | 18 | x | x | x | x | x | x | pt ₂ | pt ₁ |
| po | 19 | po ₈ | po ₇ | po ₆ | po ₅ | po ₄ | po ₃ | po ₂ | po ₁ |
| c | 20 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 21 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

7.4.3.1.5.2

Bursts defined for TIS-B tests

Table 7.78: TIS_MAN_a (TMA): Management message with response reservation field

| 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 |
| udid, ucd | 5 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| TIS-B message ID (tmi) version number (vers) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | tmi ₄ | tmi ₃ | tmi ₂ | tmi ₁ |
| TIV ID (tivid) update period (up) | 7 | up ₅ | up ₄ | up ₃ | up ₂ | up ₁ | tivid ₃ | tivid ₂ | tivid ₁ |
| TIV status (tivs) TIS-B target resolution (tacc) | 8 | lat ₁ | tacc ₄ | tacc ₃ | tacc ₂ | tacc ₁ | tivs ₂ | tivs ₁ | up ₆ |
| reference point latitude (lat) | 9 | lat ₉ | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ |
| reference point longitude (long) | 10 | long ₃ | long ₂ | long ₁ | lat ₁₄ | lat ₁₃ | lat ₁₂ | lat ₁₁ | lat ₁₀ |
| | 11 | long ₁₁ | long ₁₀ | long ₉ | long ₈ | long ₇ | long ₆ | long ₅ | long ₄ |
| lower barometric altitude (lbar) | 12 | lbar ₄ | lbar ₃ | lbar ₂ | lbar ₁ | long ₁₅ | long ₁₄ | long ₁₃ | long ₁₂ |
| upper barometric altitude (ubar) | 13 | ubar ₄ | ubar ₃ | ubar ₂ | ubar ₁ | lbar ₈ | lbar ₇ | lbar ₆ | lbar ₅ |
| number of vertices (nvert) | 14 | nvert ₄ | nvert ₃ | nvert ₂ | nvert ₁ | ubar ₈ | ubar ₇ | ubar ₆ | ubar ₅ |
| vertex latitude (vlat(1-k)) | 15 | vlat ₈ | vlat ₇ | vlat ₆ | vlat ₅ | vlat ₄ | vlat ₃ | vlat ₂ | vlat ₁ |
| vertex longitude (vlg(1-k)) | 16 | vlg ₈ | vlg ₇ | vlg ₆ | vlg ₅ | vlg ₄ | vlg ₃ | vlg ₂ | vlg ₁ |
| | | | | | | | | | |
| | m-4 | vlat ₈ | vlat ₇ | vlat ₆ | vlat ₅ | vlat ₄ | vlat ₃ | vlat ₂ | vlat ₁ |
| | m-3 | vlg ₈ | vlg ₇ | vlg ₆ | vlg ₅ | vlg ₄ | vlg ₃ | vlg ₂ | vlg ₁ |
| response reservation | m-2 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | m-1 | C ₁₆ | C ₁₅ | C ₁₄ | C ₁₃ | C ₁₂ | C ₁₁ | C ₁₀ | C ₉ |
| c | m | C ₈ | C ₇ | C ₆ | C ₅ | C ₄ | C ₃ | C ₂ | C ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

**Table 7.79: TIS_AC_AIR_a (TAa): Aircraft target message (airborne TIV)
with response reservation field**

| 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 |
| udid, ucd | 5 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| TIS-B message ID (tmi) TIV ID (tivid) target identifier flag (tflg) | 6 | tflg | tivid ₃ | tivid ₂ | tivid ₁ | tmi ₄ | tmi ₃ | tmi ₂ | tmi ₁ |
| radar/fusion flag (rflg) ADS-B fault flag (adsff) flight ID flag (fidflg) target identifier (tid) | 7 | tid ₅ | tid ₄ | tid ₃ | tid ₂ | tid ₁ | fidflg | adsff | rflg |
| | 8 | tid ₁₃ | tid ₁₂ | tid ₁₁ | tid ₁₀ | tid ₉ | tid ₈ | tid ₇ | tid ₆ |
| | 9 | tid ₂₁ | tid ₂₀ | tid ₁₉ | tid ₁₈ | tid ₁₇ | tid ₁₆ | tid ₁₅ | tid ₁₄ |
| latitude (rlat) | 10 | rlat ₅ | rlat ₄ | rlat ₃ | rlat ₂ | rlat ₁ | tid ₂₄ | tid ₂₃ | tid ₂₂ |
| | 11 | rlat ₁₃ | rlat ₁₂ | rlat ₁₁ | rlat ₁₀ | rlat ₉ | rlat ₈ | rlat ₇ | rlat ₆ |
| longitude (rlg) | 12 | rlg ₅ | rlg ₄ | rlg ₃ | rlg ₂ | rlg ₁ | rlat ₁₆ | rlat ₁₅ | rlat ₁₄ |
| | 13 | rlg ₁₃ | rlg ₁₂ | rlg ₁₁ | rlg ₁₀ | rlg ₉ | rlg ₈ | rlg ₇ | rlg ₆ |
| ground track (gtk) | 14 | gtk ₅ | gtk ₄ | gtk ₃ | gtk ₂ | gtk ₁ | rlg ₁₆ | rlg ₁₅ | rlg ₁₄ |
| barometric altitude (bar) | 15 | bar ₅ | bar ₄ | bar ₃ | bar ₂ | bar ₁ | gtk ₈ | gtk ₇ | gtk ₆ |
| ground speed (gsp) | 16 | gsp ₁ | bar ₁₂ | bar ₁₁ | bar ₁₀ | bar ₉ | bar ₈ | bar ₇ | bar ₆ |
| | 17 | gsp ₉ | gsp ₈ | gsp ₇ | gsp ₆ | gsp ₅ | gsp ₄ | gsp ₃ | gsp ₂ |
| time stamp (tag) | 18 | tag ₆ | tag ₅ | tag ₄ | tag ₃ | tag ₂ | tag ₁ | gsp ₁₁ | gsp ₁₀ |
| flight ID type (fidty) | 19 | call ₇ | call ₆ | call ₅ | call ₄ | call ₃ | call ₂ | call ₁ | fidty |
| callsign (call)(example) | 20 | call ₁₅ | call ₁₄ | call ₁₃ | call ₁₂ | call ₁₁ | call ₁₀ | call ₉ | call ₈ |
| | 21 | call ₂₃ | call ₂₂ | call ₂₁ | call ₂₀ | call ₁₉ | call ₁₈ | call ₁₇ | call ₁₆ |
| | 22 | call ₃₁ | call ₃₀ | call ₂₉ | call ₂₈ | call ₂₇ | call ₂₆ | call ₂₅ | call ₂₄ |
| | 23 | call ₃₉ | call ₃₈ | call ₃₇ | call ₃₆ | call ₃₅ | call ₃₄ | call ₃₃ | call ₃₂ |
| aircraft category (cat) | 24 | cat ₅ | cat ₄ | cat ₃ | cat ₂ | cat ₁ | call ₄₂ | call ₄₁ | call ₄₀ |
| response reservation | 25 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 26 | C ₁₆ | C ₁₅ | C ₁₄ | C ₁₃ | C ₁₂ | C ₁₁ | C ₁₀ | C ₉ |
| c | 27 | C ₈ | C ₇ | C ₆ | C ₅ | C ₄ | C ₃ | C ₂ | C ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

**Table 7.80: TIS_AC_GRD_a (TGa): Aircraft target message (ground TIV)
with response reservation field**

| 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 |
| udid, ucd | 5 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| TIS-B message ID (tmi) TIV ID (tivid) target identifier flag (tflg) | 6 | tflg | tivid ₃ | tivid ₂ | tivid ₁ | tmi ₄ | tmi ₃ | tmi ₂ | tmi ₁ |
| radar/fusion flag (rflg) ADS-B fault flag (adsff) flight ID flag (fidflg) target identifier (tid) | 7 | tid ₅ | tid ₄ | tid ₃ | tid ₂ | tid ₁ | fidflg | adsff | rflg |
| | 8 | tid ₁₃ | tid ₁₂ | tid ₁₁ | tid ₁₀ | tid ₉ | tid ₈ | tid ₇ | tid ₆ |
| | 9 | tid ₂₁ | tid ₂₀ | tid ₁₉ | tid ₁₈ | tid ₁₇ | tid ₁₆ | tid ₁₅ | tid ₁₄ |
| latitude (rlat) | 10 | rlat ₅ | rlat ₄ | rlat ₃ | rlat ₂ | rlat ₁ | tid ₂₄ | tid ₂₃ | tid ₂₂ |
| | 11 | rlat ₁₃ | rlat ₁₂ | rlat ₁₁ | rlat ₁₀ | rlat ₉ | rlat ₈ | rlat ₇ | rlat ₆ |
| longitude (rlg) | 12 | rlg ₁ | rlat ₂₀ | rlat ₁₉ | rlat ₁₈ | rlat ₁₇ | rlat ₁₆ | rlat ₁₅ | rlat ₁₄ |
| | 13 | rlg ₉ | rlg ₈ | rlg ₇ | rlg ₆ | rlg ₅ | rlg ₄ | rlg ₃ | rlg ₂ |
| | 14 | rlg ₁₇ | rlg ₁₆ | rlg ₁₅ | rlg ₁₄ | rlg ₁₃ | rlg ₁₂ | rlg ₁₁ | rlg ₁₀ |
| ground track (gtk) | 15 | gtk ₅ | gtk ₄ | gtk ₃ | gtk ₂ | gtk ₁ | rlg ₂₀ | rlg ₁₉ | rlg ₁₈ |
| ground speed (gsp) | 16 | gsp ₅ | gsp ₄ | gsp ₃ | gsp ₂ | gsp ₁ | gtk ₈ | gtk ₇ | gtk ₆ |
| time stamp (tag) | 17 | tag ₂ | tag ₁ | gsp ₁₁ | gsp ₁₀ | gsp ₉ | gsp ₈ | gsp ₇ | gsp ₆ |
| flight ID type (fidty) | 18 | res | res | res | res | tag ₆ | tag ₅ | tag ₄ | tag ₃ |
| callsign (call) | 19 | call ₇ | call ₆ | call ₅ | call ₄ | call ₃ | call ₂ | call ₁ | fidty |
| | 20 | call ₁₅ | call ₁₄ | call ₁₃ | call ₁₂ | call ₁₁ | call ₁₀ | call ₉ | call ₈ |
| | 21 | call ₂₃ | call ₂₂ | call ₂₁ | call ₂₀ | call ₁₉ | call ₁₈ | call ₁₇ | call ₁₆ |
| | 22 | call ₃₁ | call ₃₀ | call ₂₉ | call ₂₈ | call ₂₇ | call ₂₆ | call ₂₅ | call ₂₄ |
| | 23 | call ₃₉ | call ₃₈ | call ₃₇ | call ₃₆ | call ₃₅ | call ₃₄ | call ₃₃ | call ₃₂ |
| aircraft category (cat) | 24 | cat ₅ | cat ₄ | cat ₃ | cat ₂ | cat ₁ | call ₄₂ | call ₄₁ | call ₄₀ |
| response reservation | 25 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 26 | C ₁₆ | C ₁₅ | C ₁₄ | C ₁₃ | C ₁₂ | C ₁₁ | C ₁₀ | C ₉ |
| c | 27 | C ₈ | C ₇ | C ₆ | C ₅ | C ₄ | C ₃ | C ₂ | C ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.81: TIS_GRD_VEH_a (TVa): Ground vehicle target message with response reservation field

| 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 |
| udid, ucd | 5 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| TIS-B message ID (tmi) TIV ID (tivid) | 6 | tid ₁ | tivid ₃ | tivid ₂ | tivid ₁ | tmi ₄ | tmi ₃ | tmi ₂ | tmi ₁ |
| target identifier (tid) | 7 | tid ₉ | tid ₈ | tid ₇ | tid ₆ | tid ₅ | tid ₄ | tid ₃ | tid ₂ |
| ADS-B fault flag (adsff) | 8 | tag ₅ | tag ₄ | tag ₃ | tag ₂ | tag ₁ | adsff | tid ₁₁ | tid ₁₀ |
| time stamp(tag) latitude (rlat) | 9 | rlat ₇ | rlat ₆ | rlat ₅ | rlat ₄ | rlat ₃ | rlat ₂ | rlat ₁ | tag ₆ |
| | 10 | rlat ₁₅ | rlat ₁₄ | rlat ₁₃ | rlat ₁₂ | rlat ₁₁ | rlat ₁₀ | rlat ₉ | rlat ₈ |
| longitude (rlg) | 11 | rlg ₃ | rlg ₂ | rlg ₁ | rlat ₂₀ | rlat ₁₉ | rlat ₁₈ | rlat ₁₇ | rlat ₁₆ |
| | 12 | rlg ₁₁ | rlg ₁₀ | rlg ₉ | rlg ₈ | rlg ₇ | rlg ₆ | rlg ₅ | rlg ₄ |
| | 13 | rlg ₁₉ | rlg ₁₈ | rlg ₁₇ | rlg ₁₆ | rlg ₁₅ | rlg ₁₄ | rlg ₁₃ | rlg ₁₂ |
| ground track (gtk) | 14 | gtk ₇ | gtk ₆ | gtk ₅ | gtk ₄ | gtk ₃ | gtk ₂ | gtk ₁ | rlg ₂₀ |
| ground speed (gsp) | 15 | gsp ₇ | gsp ₆ | gsp ₅ | gsp ₄ | gsp ₃ | gsp ₂ | gsp ₁ | gtk ₈ |
| response reservation | 16 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| C | 18 | C ₁₆ | C ₁₅ | C ₁₄ | C ₁₃ | C ₁₂ | C ₁₁ | C ₁₀ | C ₉ |
| C | 19 | C ₈ | C ₇ | C ₆ | C ₅ | C ₄ | C ₃ | C ₂ | C ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

7.4.3.1.5.3

Bursts defined for FIS-B tests

Table 7.82: FIS_METAR_a (FMa): METAR message with response reservation field

| 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 |
| mi | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) FIS-B version ID (vers) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | 0 | 0 | 0 | 0 |
| message edition (med) local ID (locl) | 7 | locl ₅ | locl ₄ | locl ₃ | locl ₂ | locl ₁ | med ₃ | med ₂ | med ₁ |
| | 8 | locl ₁₃ | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ |
| date (date) | 9 | date ₁ | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ |
| time (ti) | 10 | ti ₄ | ti ₃ | ti ₂ | ti ₁ | date ₅ | date ₄ | date ₃ | date ₂ |
| wind direction (wdi) | 11 | wdi ₁ | ti ₁₁ | ti ₁₀ | ti ₉ | ti ₈ | ti ₇ | ti ₆ | ti ₅ |
| wind speed (wsp) | 12 | wsp ₃ | wsp ₂ | wsp ₁ | wdi ₆ | wdi ₅ | wdi ₄ | wdi ₃ | wdi ₂ |
| Free text flag (ftxt) max wind flag (mxw) CAVOK flag (cav) TREND report flag (trd) | 13 | trd | cav | mxw | ftxt | wsp ₇ | wsp ₆ | wsp ₅ | wsp ₄ |
| Present weather flag (pwf) Recent weather flag (rwf) QFE flag (qfef) temperature (tem) | 14 | tem ₃ | tem ₂ | tem ₁ | qfef | rwf ₂ | rwf ₁ | pwf ₂ | pwf ₁ |
| dew point temperature (dew) | 15 | dew ₃ | dew ₂ | dew ₁ | tem ₈ | tem ₇ | tem ₆ | tem ₅ | tem ₄ |
| QNH (qnh) | 16 | qnh ₃ | qnh ₂ | qnh ₁ | dew ₈ | dew ₇ | dew ₆ | dew ₅ | dew ₄ |

| Description | Octet | Bit number | | | | | | | |
|--|-------|-------------------|--------------------|--------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| Wind shear flag (wsf) Sea flag (ssf) variable wind direction to (vdit) | 17 | vdit ₁ | ssf | wsf | qnh ₈ | qnh ₇ | qnh ₆ | qnh ₅ | qnh ₄ |
| variable wind direction from (vdif) | 18 | vdif ₃ | vdif ₂ | vdif ₁ | vdit ₆ | vdit ₅ | vdit ₄ | vdit ₃ | vdit ₂ |
| wind gust indicator (wgi) | 19 | wgi ₅ | wgi ₄ | wgi ₃ | wgi ₂ | wgi ₁ | vdif ₆ | vdif ₅ | vdif ₄ |
| Minimum visibility (mis) | 20 | mis ₆ | mis ₅ | mis ₄ | mis ₃ | mis ₂ | mis ₁ | wgi ₇ | wgi ₆ |
| Minimum visibility direction (misd) Maximum visibility (mas) | 21 | mas ₅ | mas ₄ | mas ₃ | mas ₂ | mas ₁ | misd ₃ | misd ₂ | misd ₁ |
| Maximum visibility direction (masd) present weather (pw) | 22 | pw ₄ | pw ₃ | pw ₂ | pw ₁ | masd ₃ | masd ₂ | masd ₁ | mas ₆ |
| cloud no. flag (cno) cloud coverage (cc) | 23 | cc ₃ | cc ₂ | cc ₁ | cno ₂ | cno ₁ | pw ₇ | pw ₆ | pw ₅ |
| cloud height (ch) | 24 | ch ₈ | ch ₇ | ch ₆ | ch ₅ | ch ₄ | ch ₃ | ch ₂ | ch ₁ |
| cloud type (cty) QFE (qfe) | 25 | qfe ₆ | qfe ₅ | qfe ₄ | qfe ₃ | qfe ₂ | qfe ₁ | cty ₂ | cty ₁ |
| recent weather (rw) | 26 | rw ₆ | rw ₅ | rw ₄ | rw ₃ | rw ₂ | rw ₁ | qfe ₈ | qfe ₇ |
| wind shear (ws) | 27 | ws ₇ | ws ₆ | ws ₅ | ws ₄ | ws ₃ | ws ₂ | ws ₁ | rw ₇ |
| sea-surface temperature (sst) state of the sea (ss) | 29 | ss ₁ | sst ₆ | sst ₅ | sst ₄ | sst ₃ | sst ₂ | sst ₁ | ws ₈ |
| free text message (txt) | 30 | txt ₅ | txt ₄ | txt ₃ | txt ₂ | txt ₁ | ss ₄ | ss ₃ | ss ₂ |
| | 31 | txt _j | txt _{j-1} | txt _{j-2} | | txt ₉ | txt ₈ | txt ₇ | txt ₆ |
| response reservation | n-2 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | n-1 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | n | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.83: FIS_SPECI_a (FSa): SPECI message with response reservation field

| 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 |
| mi | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) FIS-B version ID (vers) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | 0 | 1 | 0 | 0 |
| message edition (med) local ID (locl) | 7 | locl ₅ | locl ₄ | locl ₃ | locl ₂ | locl ₁ | med ₃ | med ₂ | med ₁ |
| | 8 | locl ₁₃ | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ |
| date (date) | 9 | date ₁ | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ |
| time (ti) | 10 | ti ₄ | ti ₃ | ti ₂ | ti ₁ | date ₅ | date ₄ | date ₃ | date ₂ |
| wind direction (wdi) | 11 | wdi ₁ | ti ₁₁ | ti ₁₀ | ti ₉ | ti ₈ | ti ₇ | ti ₆ | ti ₅ |
| wind speed (wsp) | 12 | wsp ₃ | wsp ₂ | wsp ₁ | wdi ₆ | wdi ₅ | wdi ₄ | wdi ₃ | wdi ₂ |
| Free text flag (ftxt) max wind flag (mxw) CAVOK flag (cav) TREND report flag (trd) | 13 | trd | cav | mxw | ftxt | wsp ₇ | wsp ₆ | wsp ₅ | wsp ₄ |
| Present weather flag (pwf) Recent weather flag (rwf) QFE flag (qfef) temperature (tem) | 14 | tem ₃ | tem ₂ | tem ₁ | qfef | rwf ₂ | rwf ₁ | pwf ₂ | pwf ₁ |
| dew point temperature (dew) | 15 | dew ₃ | dew ₂ | dew ₁ | tem ₈ | tem ₇ | tem ₆ | tem ₅ | tem ₄ |
| QNH (qnh) | 16 | qnh ₃ | qnh ₂ | qnh ₁ | dew ₈ | dew ₇ | dew ₆ | dew ₅ | dew ₄ |

| Description | Octet | Bit number | | | | | | | |
|--|-------|-------------------|--------------------|--------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| Wind shear flag (wsf) Sea flag (ssf) variable wind direction to (vdit) | 17 | vdi _{t1} | ssf | wsf | qnh ₈ | qnh ₇ | qnh ₆ | qnh ₅ | qnh ₄ |
| variable wind direction from (vdif) | 18 | vdif ₃ | vdif ₂ | vdif ₁ | vdi _{t6} | vdi _{t5} | vdi _{t4} | vdi _{t3} | vdi _{t2} |
| wind gust indicator (wgi) | 19 | wgi ₅ | wgi ₄ | wgi ₃ | wgi ₂ | wgi ₁ | vdif ₆ | vdif ₅ | vdif ₄ |
| Minimum visibility (mis) | 20 | mis ₆ | mis ₅ | mis ₄ | mis ₃ | mis ₂ | mis ₁ | wgi ₇ | wgi ₆ |
| Minimum visibility direction (misd) Maximum visibility (mas) | 21 | mas ₅ | mas ₄ | mas ₃ | mas ₂ | mas ₁ | misd ₃ | misd ₂ | misd ₁ |
| Maximum visibility direction (masd) present weather (pw) | 22 | pw ₄ | pw ₃ | pw ₂ | pw ₁ | masd ₃ | masd ₂ | masd ₁ | mas ₆ |
| cloud no. flag (cno) cloud coverage (cc) | 23 | cc ₃ | cc ₂ | cc ₁ | cno ₂ | cno ₁ | pw ₇ | pw ₆ | pw ₅ |
| cloud height (ch) | 24 | ch ₈ | ch ₇ | ch ₆ | ch ₅ | ch ₄ | ch ₃ | ch ₂ | ch ₁ |
| cloud type (cty) QFE (qfe) | 25 | qfe ₆ | qfe ₅ | qfe ₄ | qfe ₃ | qfe ₂ | qfe ₁ | cty ₂ | cty ₁ |
| recent weather (rw) | 26 | rw ₆ | rw ₅ | rw ₄ | rw ₃ | rw ₂ | rw ₁ | qfe ₈ | qfe ₇ |
| wind shear (ws) | 27 | ws ₇ | ws ₆ | ws ₅ | ws ₄ | ws ₃ | ws ₂ | ws ₁ | rw ₇ |
| sea-surface temperature (sst) state of the sea (ss) | 29 | ss ₁ | sst ₆ | sst ₅ | sst ₄ | sst ₃ | sst ₂ | sst ₁ | ws ₈ |
| free text message (txt) | 30 | txt ₅ | txt ₄ | txt ₃ | txt ₂ | txt ₁ | ss ₄ | ss ₃ | ss ₂ |
| | 31 | txt _j | txt _{j-1} | txt _{j-2} | | txt ₉ | txt ₈ | txt ₇ | txt ₆ |
| response reservation | n-2 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | n-1 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | n | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.84: FIS_ATIS_a (FAa): ATIS message with response reservation field

| 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 |
| mi | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) FIS-B version ID (vers) message edition (med) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | 0 | 0 | 0 | 1 |
| local ID(locl) | 7 | locl ₅ | locl ₄ | locl ₃ | locl ₂ | locl ₁ | med ₃ | med ₂ | med ₁ |
| | 8 | locl ₁₃ | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ |
| date (date) | 9 | date ₁ | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ |
| time (ti) | 10 | ti ₄ | ti ₃ | ti ₂ | ti ₁ | date ₅ | date ₄ | date ₃ | date ₂ |
| holding delay (hd) | 11 | hd ₁ | ti ₁₁ | ti ₁₀ | ti ₉ | ti ₈ | ti ₇ | ti ₆ | ti ₅ |
| transition level (tran) | 12 | tran ₅ | tran ₄ | tran ₃ | tran ₂ | tran ₁ | hd ₄ | hd ₃ | hd ₂ |
| other op. activity flag (othf) other op. activity (oth) airport availability (aavl) | 13 | aavl ₃ | aavl ₂ | aavl ₁ | oth ₃ | oth ₂ | oth ₁ | othf | tran ₆ |
| | 14 | aavl ₁₁ | aavl ₁₀ | aavl ₉ | aavl ₈ | aavl ₇ | aavl ₆ | aavl ₅ | aavl ₄ |
| wind direction (wdi) wind speed (wsp) | 15 | wsp ₁ | wdi ₆ | wdi ₅ | wdi ₄ | wdi ₃ | wdi ₂ | wdi ₁ | aavl ₁₂ |
| temperature (tem) | 16 | tem ₂ | tem ₁ | wsp ₇ | wsp ₆ | wsp ₅ | wsp ₄ | wsp ₃ | wsp ₂ |
| dew point (dew) | 17 | dew ₂ | dew ₁ | tem ₈ | tem ₇ | tem ₆ | tem ₅ | tem ₄ | tem ₃ |
| QNH (qnh) | 18 | qnh ₂ | qnh ₁ | dew ₈ | dew ₇ | dew ₆ | dew ₅ | dew ₄ | dew ₃ |

| Description | Octet | Bit number | | | | | | | |
|--|-------|-------------------|--------------------|--------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| free text flag (ftxt) max wind flag (mxw) CAVOK flag (cav) | 19 | mxw | ftxt | qnh ₈ | qnh ₇ | qnh ₆ | qnh ₅ | qnh ₄ | qnh ₃ |
| TREND report flag (trd) QFE flag (qfef) wind shear flag (ws) Present weather flag (pwf) Recent weather flag (rwf) variable wind direction to (vdit) | 20 | rwf ₂ | rwf ₁ | pwf ₂ | pwf ₁ | ws | qfef | trd | cav |
| variable wind direction from (vdif) | 21 | vdif ₂ | vdif ₁ | vdit ₆ | vdit ₅ | vdit ₄ | vdit ₃ | vdit ₂ | vdit ₁ |
| wind gust indicator (wgi) | 22 | wgi ₄ | wgi ₃ | wgi ₂ | wgi ₁ | vdif ₆ | vdif ₅ | vdif ₄ | vdif ₃ |
| Minimum visibility (mis) | 23 | mis ₅ | mis ₄ | mis ₃ | mis ₂ | mis ₁ | wgi ₇ | wgi ₆ | wgi ₅ |
| Minimum visibility direction (misd) Maximum visibility (mas) | 24 | mas ₄ | mas ₃ | mas ₂ | mas ₁ | misd ₃ | misd ₂ | misd ₁ | mis ₆ |
| Maximum visibility direction (masd) cloud no. flag (cno) cloud coverage (cc) | 25 | cc ₁ | cno ₂ | cno ₁ | masd ₃ | masd ₂ | masd ₁ | mas ₆ | mas ₅ |
| cloud height (ch) | 26 | ch ₆ | ch ₅ | ch ₄ | ch ₃ | ch ₂ | ch ₁ | cc ₃ | cc ₂ |
| cloud type (cty) present weather (pw) | 27 | pw ₄ | pw ₃ | pw ₂ | pw ₁ | cty ₂ | cty ₁ | ch ₈ | ch ₇ |
| recent weather (rw) | 28 | rw ₅ | rw ₄ | rw ₃ | rw ₂ | rw ₁ | pw ₇ | pw ₆ | pw ₅ |
| QFE (qfe) | 29 | qfe ₆ | qfe ₅ | qfe ₄ | qfe ₃ | qfe ₂ | qfe ₁ | rw ₇ | rw ₆ |
| free text message (txt) | 30 | txt ₆ | txt ₅ | txt ₄ | txt ₃ | txt ₂ | txt ₁ | qfe ₈ | qfe ₇ |
| | 31 | txt _j | txt _{j-1} | txt _{j-2} | | txt ₁₀ | txt ₉ | txt ₈ | txt ₇ |
| response reservation field | n-2 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | n-1 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | n | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.85: FIS_RCN_a (FRa): RCN message with response reservation field

| 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 |
| mi | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) FIS-B version ID (vers) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | 0 | 0 | 1 | 0 |
| message edition (med) local ID (locl) | 7 | locl ₅ | locl ₄ | locl ₃ | locl ₂ | locl ₁ | med ₃ | med ₂ | med ₁ |
| | 8 | locl ₁₃ | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ |
| date (date) | 9 | date ₁ | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ |
| time (ti) | 10 | ti ₄ | ti ₃ | ti ₂ | ti ₁ | date ₅ | date ₄ | date ₃ | date ₂ |
| braking action (brk) | 11 | brk ₁ | ti ₁₁ | ti ₁₀ | ti ₉ | ti ₈ | ti ₇ | ti ₆ | ti ₅ |
| runway ID (run) | 12 | run ₆ | run ₅ | run ₄ | run ₃ | run ₂ | run ₁ | brk ₃ | brk ₂ |
| Approach type flag (aptf) Approach type (apty) | 13 | apty ₄ | apty ₃ | apty ₂ | apty ₁ | aptf ₂ | aptf ₁ | run ₈ | run ₇ |
| reduced runway length (rrl) | 14 | rrl ₈ | rrl ₇ | rrl ₆ | rrl ₅ | rrl ₄ | rrl ₃ | rrl ₂ | rrl ₁ |
| reduced runway width (rrw) runway deposits (rdp) | 15 | rdp ₁ | rrw ₇ | rrw ₆ | rrw ₅ | rrw ₄ | rrw ₃ | rrw ₂ | rrw ₁ |
| runway contamination (rcon) depth of deposit (dod) | 16 | dod ₂ | dod ₁ | rcon ₃ | rcon ₂ | rcon ₁ | rdp ₄ | rdp ₃ | rdp ₂ |

| Description | Octet | Bit number | | | | | | | |
|---|-------|-------------------|--------------------|--------------------|--------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| RVR touchdown (rvt) | 17 | rvt ₃ | rvt ₂ | rvt ₁ | dod ₇ | dod ₆ | dod ₅ | dod ₄ | dod ₃ |
| RVR midpoint (rvm) | 18 | rvm ₅ | rvm ₄ | rvm ₃ | rvm ₂ | rvm ₁ | rvt ₆ | rvt ₅ | rvt ₄ |
| RVR endpoint (rve) RVR source (rso) | 19 | rso ₁ | rve ₆ | rve ₅ | rve ₄ | rve ₃ | rve ₂ | rve ₁ | rvm ₆ |
| runway in use (rus) runway availability (ravl) | 20 | ravl ₅ | ravl ₄ | ravl ₃ | ravl ₂ | ravl ₁ | rus ₂ | rus ₁ | rso ₂ |
| wind shear | 21 | ws | ravl ₁₂ | ravl ₁₁ | ravl ₁₀ | ravl ₉ | ravl ₈ | ravl ₇ | ravl ₆ |
| response reservation field | 22 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 23 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 24 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.86: FIS_SIGMET_a (FSa): SIGMET message with response reservation field

| 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 |
| mi | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) FIS-B sequence number (sqn) | 6 | sqn ₄ | sqn ₃ | sqn ₂ | sqn ₁ | 0 | 0 | 1 | 1 |
| message edition (med) location indicator (loci) | 7 | loci ₅ | loci ₄ | loci ₃ | loci ₂ | loci ₁ | med ₃ | med ₂ | med ₁ |
| location indicator (loci) | 8 | loci ₁₃ | loci ₁₂ | loci ₁₁ | loci ₁₀ | loci ₉ | loci ₈ | loci ₇ | loci ₆ |
| originator indicator (ori) | 9 | ori ₁ | loci ₂₀ | loci ₁₉ | loci ₁₈ | loci ₁₇ | loci ₁₆ | loci ₁₅ | loci ₁₄ |
| | 10 | ori ₉ | ori ₈ | ori ₇ | ori ₆ | ori ₅ | ori ₄ | ori ₃ | ori ₂ |
| | 11 | ori ₁₇ | ori ₁₆ | ori ₁₅ | ori ₁₄ | ori ₁₃ | ori ₁₂ | ori ₁₁ | ori ₁₀ |
| date issued (di) time issued (ti) | 12 | di ₅ | di ₄ | di ₃ | di ₂ | di ₁ | ori ₂₀ | ori ₁₉ | ori ₁₈ |
| date valid to (dvt) time valid to (tvt) | 13 | ti ₈ | ti ₇ | ti ₆ | ti ₅ | ti ₄ | ti ₃ | ti ₂ | ti ₁ |
| date valid to (dvt) time valid to (tvt) | 14 | dvt ₅ | dvt ₄ | dvt ₃ | dvt ₂ | dvt ₁ | ti ₁₁ | ti ₁₀ | ti ₉ |
| SIGMET flag (sfl) free text flag (ftxt) phenomenon / condition reported (crp) | 15 | crp ₃ | crp ₂ | crp ₁ | ftxt | sfl | tvt ₁₁ | tvt ₁₀ | tvt ₉ |
| free text message (txt) | 15 | txt ₆ | txt ₅ | txt ₄ | txt ₃ | txt ₂ | txt ₁ | crp ₅ | crp ₄ |
| | 16 | txt _j | txt _{j-1} | txt _{j-2} | | txt ₁₀ | txt ₉ | txt ₈ | txt ₇ |
| response reservation field | n-2 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | n-1 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | n | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.87: FIS_TSA_CTR_a (FTCa): TSA CTR Zone message with response reservation field

| 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 |
| mi | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) version ID (vers) message edition (med) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | 0 | 1 | 1 | 0 |
| date (date) | 7 | date ₅ | date ₄ | date ₃ | date ₂ | date ₁ | med ₃ | med ₂ | med ₁ |
| time (ti) | 8 | ti ₈ | ti ₇ | ti ₆ | ti ₅ | ti ₄ | ti ₃ | ti ₂ | ti ₁ |
| Number of TSAs (NoT) | 9 | NoT ₅ | NoT ₄ | NoT ₃ | NoT ₂ | NoT ₁ | ti ₁₁ | ti ₁₀ | ti ₉ |
| Type of Area (toa) Active flag (act) local ID (locl) | 10 | locl ₄ | locl ₃ | locl ₂ | locl ₁ | act ₁ | toa ₃ | toa ₂ | toa ₁ |
| | 11 | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ | locl ₅ |
| | 12 | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ | locl ₁₃ |
| supplement to local ID (sloc) active from/to date (actdat) | 13 | acdat ₃ | acdat ₂ | acdat ₁ | sloc ₅ | sloc ₄ | sloc ₃ | sloc ₂ | sloc ₁ |
| active from/to time (actim) | 14 | actim ₆ | actim ₅ | actim ₄ | actim ₃ | actim ₂ | actim ₁ | acdat ₅ | acdat ₄ |
| TWR frequency (freq) | 15 | freq ₃ | freq ₂ | freq ₁ | actim ₁₁ | actim ₁₀ | actim ₉ | actim ₈ | actim ₇ |
| | 16 | freq ₁₁ | freq ₁₀ | freq ₉ | freq ₈ | freq ₇ | freq ₆ | freq ₅ | freq ₄ |
| response reservation field | n-2 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | n-1 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | n | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.88: FIS_TSA_DNG_a(FTDa): TSA DNG Zone message with response reservation field

| 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 |
| mi | 5 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| FIS-B message ID (fmi) version ID (vers) message edition (med) | 6 | vers ₄ | vers ₃ | vers ₂ | vers ₁ | 0 | 1 | 1 | 0 |
| date (date) | 7 | date ₅ | date ₄ | date ₃ | date ₂ | date ₁ | med ₃ | med ₂ | med ₁ |
| time (ti) | 8 | ti ₈ | ti ₇ | ti ₆ | ti ₅ | ti ₄ | ti ₃ | ti ₂ | ti ₁ |
| Number of TSAs (NoT) | 9 | NoT ₅ | NoT ₄ | NoT ₃ | NoT ₂ | NoT ₁ | ti ₁₁ | ti ₁₀ | ti ₉ |
| Type of Area (toa) Active flag (act) state ID (stid) | 10 | stid ₄ | stid ₃ | stid ₂ | stid ₁ | act ₁ | toa ₃ | toa ₂ | toa ₁ |
| Area leap number (aln) | 11 | aln ₂ | aln ₁ | stid ₁₀ | stid ₉ | stid ₈ | stid ₇ | stid ₆ | stid ₅ |
| | 12 | aln ₁₀ | aln ₉ | aln ₈ | aln ₇ | aln ₆ | aln ₅ | aln ₄ | aln ₃ |
| Supplement to area leap number (saln) | 13 | saln ₅ | saln ₄ | saln ₃ | saln ₂ | saln ₁ | aln ₁₃ | aln ₁₂ | aln ₁₁ |
| active from/to date (actdat) active from/to time (actim) | 14 | actim ₃ | actim ₂ | actim ₁ | acdat ₅ | acdat ₄ | acdat ₃ | acdat ₂ | acdat ₁ |
| | 15 | actim ₁₁ | actim ₁₀ | actim ₉ | actim ₈ | actim ₇ | actim ₆ | actim ₅ | actim ₄ |
| response reservation field | n-2 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | n-1 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | n | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.89: FIS_REQUEST_a (FQa): Request message with null reservation field

| 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 | 1 | 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 |
| mi = general request burst r-mi = FIS-B service indicator | 5 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| FIS-B message ID (fmi) message requested (mrq) | 6 | mrq ₁ | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| local ID (locl) | 7 | locl ₅ | locl ₄ | locl ₃ | locl ₂ | locl ₁ | mrq ₄ | mrq ₃ | mrq ₂ |
| | 8 | locl ₁₃ | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ |
| res | 9 | res | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ |
| res, null res field | 10 | res | res | res | res | res | res | 0 | 0 |
| null res field | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| c | 12 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 13 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.90: FIS_REQUEST_b (FQb): Request message with unicast reservation field

| Description | Octet | Bit number | | | | | | | | |
|--|-------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| s, ver, rid, a/d | 1 | s ₂₇ | s ₂₆ | s ₂₅ | 0 | 0 | 0 | 0 | 1 | 1 |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ | s ₁₆ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ | s ₈ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ | s ₀ |
| mi = general request burst r-mi = FIS-B service indicator | 5 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| FIS-B message ID (fmi) message requested (mrq) | 6 | mrq ₁ | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| local ID (locl) | 7 | locl ₅ | locl ₄ | locl ₃ | locl ₂ | locl ₁ | mrq ₄ | mrq ₃ | mrq ₂ | mrq ₁ |
| | 8 | locl ₁₃ | locl ₁₂ | locl ₁₁ | locl ₁₀ | locl ₉ | locl ₈ | locl ₇ | locl ₆ | locl ₅ |
| res | 9 | res | locl ₂₀ | locl ₁₉ | locl ₁₈ | locl ₁₇ | locl ₁₆ | locl ₁₅ | locl ₁₄ | locl ₁₃ |
| d | 10 | d ₂₄ | d ₂₃ | d ₂₂ | d ₂₁ | d ₂₀ | d ₁₉ | d ₁₈ | d ₁₇ | d ₁₆ |
| d | 11 | d ₁₆ | d ₁₅ | d ₁₄ | d ₁₃ | d ₁₂ | d ₁₁ | d ₁₀ | d ₉ | d ₈ |
| d | 12 | d ₈ | d ₇ | d ₆ | d ₅ | d ₄ | d ₃ | d ₂ | d ₁ | d ₀ |
| ro, sdf, d | 13 | ro ₁₂ | ro ₁₁ | ro ₁₀ | ro ₉ | sdf | d ₂₇ | d ₂₆ | d ₂₅ | d ₂₄ |
| ro | 14 | ro ₈ | ro ₇ | ro ₆ | ro ₅ | ro ₄ | ro ₃ | ro ₂ | ro ₁ | ro ₀ |
| lg | 15 | lg ₈ | lg ₇ | lg ₆ | lg ₅ | lg ₄ | lg ₃ | lg ₂ | lg ₁ | lg ₀ |
| pr | 16 | 0 | 0 | 1 | 0 | pr ₄ | pr ₃ | pr ₂ | pr ₁ | pr ₀ |
| c | 17 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ | c ₈ |
| c | 18 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ | c ₀ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

7.4.3.1.5.4

Bursts defined for GNS-B tests

Table 7.91: GNS_TYPE1_a (G1a): GNS-B Type 1 message with response reservation

| Description | Octet | Bit number | | | | | | | | |
|--|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| s, ver, rid, a/d | 1 | s ₂₇ | s ₂₆ | s ₂₅ | 0 | 0 | 0 | 0 | 0 | 1 |
| s | 2 | s ₂₄ | s ₂₃ | s ₂₂ | s ₂₁ | s ₂₀ | s ₁₉ | s ₁₈ | s ₁₇ | s ₁₆ |
| s | 3 | s ₁₆ | s ₁₅ | s ₁₄ | s ₁₃ | s ₁₂ | s ₁₁ | s ₁₀ | s ₉ | s ₈ |
| s | 4 | s ₈ | s ₇ | s ₆ | s ₅ | s ₄ | s ₃ | s ₂ | s ₁ | s ₀ |
| udid, ucd, mi | 5 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| message block identifier (mbi) | 6 | mbi ₈ | mbi ₇ | mbi ₆ | mbi ₅ | mbi ₄ | mbi ₃ | mbi ₂ | mbi ₁ | mbi ₀ |
| GNS-B ID (gid) | 7 | gid ₈ | gid ₇ | gid ₆ | gid ₅ | gid ₄ | gid ₃ | gid ₂ | gid ₁ | gid ₀ |
| | 8 | gid ₁₆ | gid ₁₅ | gid ₁₄ | gid ₁₃ | gid ₁₂ | gid ₁₁ | gid ₁₀ | gid ₉ | gid ₈ |
| | 9 | gid ₂₄ | gid ₂₃ | gid ₂₂ | gid ₂₁ | gid ₂₀ | gid ₁₉ | gid ₁₈ | gid ₁₇ | gid ₁₆ |
| message identifier (gmi) | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| message length (len) | 11 | len ₈ | len ₇ | len ₆ | len ₅ | len ₄ | len ₃ | len ₂ | len ₁ | len ₀ |
| modified Z-count (zc) | 12 | zc ₈ | zc ₇ | zc ₆ | zc ₅ | zc ₄ | zc ₃ | zc ₂ | zc ₁ | zc ₀ |
| additional message flag (amf) | 13 | amf ₂ | amf ₁ | zc ₁₄ | zc ₁₃ | zc ₁₂ | zc ₁₁ | zc ₁₀ | zc ₉ | zc ₈ |
| number of measurements (n) measurement type (t) | 14 | t ₃ | t ₂ | t ₁ | n ₅ | n ₄ | n ₃ | n ₂ | n ₁ | n ₀ |
| ephemeris decorrelation parameter (edp) | 15 | edp ₈ | edp ₇ | edp ₆ | edp ₅ | edp ₄ | edp ₃ | edp ₂ | edp ₁ | edp ₀ |
| ephemeris CRC(ec) | 16 | ec ₈ | ec ₇ | ec ₆ | ec ₅ | ec ₄ | ec ₃ | ec ₂ | ec ₁ | ec ₀ |
| | 17 | ec ₁₆ | ec ₁₅ | ec ₁₄ | ec ₁₃ | ec ₁₂ | ec ₁₁ | ec ₁₀ | ec ₉ | ec ₈ |
| source availability duration (ad) | 18 | ad ₈ | ad ₇ | ad ₆ | ad ₅ | ad ₄ | ad ₃ | ad ₂ | ad ₁ | ad ₀ |

| Description | Octet | Bit number | | | | | | | |
|---|-------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| For N measurement blocks (i=1 to i=N): ranging source ID (id) | 19+(i-1)x11 | id ₈ | id ₇ | id ₆ | id ₅ | id ₄ | id ₃ | id ₂ | id ₁ |
| issue of data (iod) | 20+(i-1)x11 | iod ₈ | iod ₇ | iod ₆ | iod ₅ | iod ₄ | iod ₃ | iod ₂ | iod ₁ |
| pseudorange correction (prc) | 21+(i-1)x11 | prc ₈ | prc ₇ | prc ₆ | prc ₅ | prc ₄ | prc ₃ | prc ₂ | prc ₁ |
| | 22+(i-1)x11 | prc ₁₆ | prc ₁₅ | prc ₁₄ | prc ₁₃ | prc ₁₂ | prc ₁₁ | prc ₁₀ | prc ₉ |
| range rate correction (rrc) | 23+(i-1)x11 | rrc ₈ | rrc ₇ | rrc ₆ | rrc ₅ | rrc ₄ | rrc ₃ | rrc ₂ | rrc ₁ |
| | 24+(i-1)x11 | rrc ₁₆ | rrc ₁₅ | rrc ₁₄ | rrc ₁₃ | rrc ₁₂ | rrc ₁₁ | rrc ₁₀ | rrc ₉ |
| σ_{pr_gnd} (sd) | 25+(i-1)x11 | sd ₈ | sd ₇ | sd ₆ | sd ₅ | sd ₄ | sd ₃ | sd ₂ | sd ₁ |
| integrity parameter B1 (b1) | 26+(i-1)x11 | b1 ₈ | b1 ₇ | b1 ₆ | b1 ₅ | b1 ₄ | b1 ₃ | b1 ₂ | b1 ₁ |
| integrity parameter B2 (b2) | 27+(i-1)x11 | b2 ₈ | b2 ₇ | b2 ₆ | b2 ₅ | b2 ₄ | b2 ₃ | b2 ₂ | b2 ₁ |
| integrity parameter B3 (b3) | 28+(i-1)x11 | b3 ₈ | b3 ₇ | b3 ₆ | b3 ₅ | b3 ₄ | b3 ₃ | b3 ₂ | b3 ₁ |
| integrity parameter B4 (b4) | 29+(i-1)x11 | b4 ₈ | b4 ₇ | b4 ₆ | b4 ₅ | b4 ₄ | b4 ₃ | b4 ₂ | b4 ₁ |
| GNS-B message CRC (gc) | 30+(N-1)x11 | gc ₈ | gc ₇ | gc ₆ | gc ₅ | gc ₄ | gc ₃ | gc ₂ | gc ₁ |
| | 31+(N-1)x11 | gc ₁₆ | gc ₁₅ | gc ₁₄ | gc ₁₃ | gc ₁₂ | gc ₁₁ | gc ₁₀ | gc ₉ |
| | 32+(N-1)x11 | gc ₂₄ | gc ₂₃ | gc ₂₂ | gc ₂₁ | gc ₂₀ | gc ₁₉ | gc ₁₈ | gc ₁₇ |
| | 33+(N-1)x11 | gc ₃₂ | gc ₃₁ | gc ₃₀ | gc ₂₉ | gc ₂₈ | gc ₂₇ | gc ₂₆ | gc ₂₅ |
| response reservation field | 34+(N-1)x11 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 35+(N-1)x11 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 36+(N-1)x11 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.92: GNS_TYPE2_a (G2a): GNS-B Type 2 message with response 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 |
| udid, ucd, mi | 5 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| message block identifier (mbi) | 6 | mbi ₈ | mbi ₇ | mbi ₆ | mbi ₅ | mbi ₄ | mbi ₃ | mbi ₂ | mbi ₁ |
| GNS-B ID (gid) | 7 | gid ₈ | gid ₇ | gid ₆ | gid ₅ | gid ₄ | gid ₃ | gid ₂ | gid ₁ |
| | 8 | gid ₁₆ | gid ₁₅ | gid ₁₄ | gid ₁₃ | gid ₁₂ | gid ₁₁ | gid ₁₀ | gid ₉ |
| | 9 | gid ₂₄ | gid ₂₃ | gid ₂₂ | gid ₂₁ | gid ₂₀ | gid ₁₉ | gid ₁₈ | gid ₁₇ |
| message identifier (gmi) | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| message length (len) | 11 | len ₈ | len ₇ | len ₆ | len ₅ | len ₄ | len ₃ | len ₂ | len ₁ |
| GNS-B reference receivers (r) GNS-B accuracy designator letter (ac) GNS-B continuity/integrity designator (cid) | 12 | cid ₃ | cid ₂ | cid ₁ | res | ac ₂ | ac ₁ | r ₂ | r ₁ |
| local magnetic variation (v) | 13 | v ₈ | v ₇ | v ₆ | v ₅ | v ₄ | v ₃ | v ₂ | v ₁ |
| res | 14 | res | res | res | res | res | v ₁₁ | v ₁₀ | v ₉ |
| σ vert iono grad (vi) | 15 | vi ₈ | vi ₇ | vi ₆ | vi ₅ | vi ₄ | vi ₃ | vi ₂ | vi ₁ |
| refractivity index (i) | 16 | i ₈ | i ₇ | i ₆ | i ₅ | i ₄ | i ₃ | i ₂ | i ₁ |
| scale height (h) | 17 | h ₈ | h ₇ | h ₆ | h ₅ | h ₄ | h ₃ | h ₂ | h ₁ |
| refractivity uncertainty (u) | 18 | u ₈ | u ₇ | u ₆ | u ₅ | u ₄ | u ₃ | u ₂ | u ₁ |

| Description | Octet | Bit number | | | | | | | |
|----------------------------|-------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| latitude (lat) | 19 | lat ₈ | lat ₇ | lat ₆ | lat ₅ | lat ₄ | lat ₃ | lat ₂ | lat ₁ |
| | 20 | lat ₁₆ | lat ₁₅ | lat ₁₄ | lat ₁₃ | lat ₁₂ | lat ₁₁ | lat ₁₀ | lat ₉ |
| | 21 | lat ₂₄ | lat ₂₃ | lat ₂₂ | lat ₂₁ | lat ₂₀ | lat ₁₉ | lat ₁₈ | lat ₁₇ |
| | 22 | lat ₃₂ | lat ₃₁ | lat ₃₀ | lat ₂₉ | lat ₂₈ | lat ₂₇ | lat ₂₆ | lat ₂₅ |
| longitude (lon) | 23 | lon ₈ | lon ₇ | lon ₆ | lon ₅ | lon ₄ | lon ₃ | lon ₂ | lon ₁ |
| | 24 | lon ₁₆ | lon ₁₅ | lon ₁₄ | lon ₁₃ | lon ₁₂ | lon ₁₁ | lon ₁₀ | lon ₉ |
| | 25 | lon ₂₄ | lon ₂₃ | lon ₂₂ | lon ₂₁ | lon ₂₀ | lon ₁₉ | lon ₁₈ | lon ₁₇ |
| | 26 | lon ₃₂ | lon ₃₁ | lon ₃₀ | lon ₂₉ | lon ₂₈ | lon ₂₇ | lon ₂₆ | lon ₂₅ |
| ellipsoid height (h) | 27 | h ₈ | h ₇ | h ₆ | h ₅ | h ₄ | h ₃ | h ₂ | h ₁ |
| | 28 | h ₁₆ | h ₁₅ | h ₁₄ | h ₁₃ | h ₁₂ | h ₁₁ | h ₁₀ | h ₉ |
| | 29 | h ₂₄ | h ₂₃ | h ₂₂ | h ₂₁ | h ₂₀ | h ₁₉ | h ₁₈ | h ₁₇ |
| GNS-B Message CRC (gc) | 30 | gc ₈ | gc ₇ | gc ₆ | gc ₅ | gc ₄ | gc ₃ | gc ₂ | gc ₁ |
| | 31 | gc ₁₆ | gc ₁₅ | gc ₁₄ | gc ₁₃ | gc ₁₂ | gc ₁₁ | gc ₁₀ | gc ₉ |
| | 32 | gc ₂₄ | gc ₂₃ | gc ₂₂ | gc ₂₁ | gc ₂₀ | gc ₁₉ | gc ₁₈ | gc ₁₇ |
| | 33 | gc ₃₂ | gc ₃₁ | gc ₃₀ | gc ₂₉ | gc ₂₈ | gc ₂₇ | gc ₂₆ | gc ₂₅ |
| response reservation field | 34 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 35 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 36 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 7.93: GNS_TYPE4_a (G4a): GNS-B Type 4 message with response 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 |
| udid, ucd, mi | 5 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| message block identifier (mbi) | 6 | mbi ₈ | mbi ₇ | mbi ₆ | mbi ₅ | mbi ₄ | mbi ₃ | mbi ₂ | mbi ₁ |
| GNS-B ID (gid) | 7 | gid ₈ | gid ₇ | gid ₆ | gid ₅ | gid ₄ | gid ₃ | gid ₂ | gid ₁ |
| | 8 | gid ₁₆ | gid ₁₅ | gid ₁₄ | gid ₁₃ | gid ₁₂ | gid ₁₁ | gid ₁₀ | gid ₉ |
| | 9 | gid ₂₄ | gid ₂₃ | gid ₂₂ | gid ₂₁ | gid ₂₀ | gid ₁₉ | gid ₁₈ | gid ₁₇ |
| message identifier (gmi) | 10 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| message length (len) | 11 | len ₈ | len ₇ | len ₆ | len ₅ | len ₄ | len ₃ | len ₂ | len ₁ |
| data set length (l) | 12 | l ₈ | l ₇ | l ₆ | l ₅ | l ₄ | l ₃ | l ₂ | l ₁ |
| For N FAS data blocks (i=1 to i=N): operation type (ot) SBAS provider ID (sp) | 13+(i-1)x40 | sp ₄ | sp ₃ | sp ₂ | sp ₁ | ot ₄ | ot ₃ | ot ₂ | ot ₁ |
| airport ID (ai) | 14+(i-1)x40 | ai ₈ | ai ₇ | ai ₆ | ai ₅ | ai ₄ | ai ₃ | ai ₂ | ai ₁ |
| | 15+(i-1)x40 | ai ₁₆ | ai ₁₅ | ai ₁₄ | ai ₁₃ | ai ₁₂ | ai ₁₁ | ai ₁₀ | ai ₉ |
| | 16+(i-1)x40 | ai ₂₄ | ai ₂₃ | ai ₂₂ | ai ₂₁ | ai ₂₀ | ai ₁₉ | ai ₁₈ | ai ₁₇ |
| | 17+(i-1)x40 | ai ₃₂ | ai ₃₁ | ai ₃₀ | ai ₂₉ | ai ₂₈ | ai ₂₇ | ai ₂₆ | ai ₂₅ |
| runway number (rn) runway letter (rl) | 18+(i-1)x40 | rl ₂ | rl ₁ | rn ₆ | rn ₅ | rn ₄ | rn ₃ | rn ₂ | rn ₁ |
| approach performance designator (apd) route indicator (ri) | 19+(i-1)x40 | ri ₅ | ri ₄ | ri ₃ | ri ₂ | ri ₁ | apd ₃ | apd ₂ | apd ₁ |
| reference path data selector (rps) | 20+(i-1)x40 | rps ₈ | rps ₇ | rps ₆ | rps ₅ | rps ₄ | rps ₃ | rps ₂ | rps ₁ |

| Description | Octet | Bit number | | | | | | | |
|---|-------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| | | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| reference path ID (rpi) | 21+(i-1)x40 | rpi ₈ | rpi ₇ | rpi ₆ | rpi ₅ | rpi ₄ | rpi ₃ | rpi ₂ | rpi ₁ |
| | 22+(i-1)x40 | rpi ₁₆ | rpi ₁₅ | rpi ₁₄ | rpi ₁₃ | rpi ₁₂ | rpi ₁₁ | rpi ₁₀ | rpi ₉ |
| | 23+(i-1)x40 | rpi ₂₄ | rpi ₂₃ | rpi ₂₂ | rpi ₂₁ | rpi ₂₀ | rpi ₁₉ | rpi ₁₈ | rpi ₁₇ |
| | 24+(i-1)x40 | rpi ₃₂ | rpi ₃₁ | rpi ₃₀ | rpi ₂₉ | rpi ₂₈ | rpi ₂₇ | rpi ₂₆ | rpi ₂₅ |
| LTP/FTP latitude (ltla) | 25+(i-1)x40 | ltla ₈ | ltla ₇ | ltla ₆ | ltla ₅ | ltla ₄ | ltla ₃ | ltla ₂ | ltla ₁ |
| | 26+(i-1)x40 | ltla ₁₆ | ltla ₁₅ | ltla ₁₄ | ltla ₁₃ | ltla ₁₂ | ltla ₁₁ | ltla ₁₀ | ltla ₉ |
| | 27+(i-1)x40 | ltla ₂₄ | ltla ₂₃ | ltla ₂₂ | ltla ₂₁ | ltla ₂₀ | ltla ₁₉ | ltla ₁₈ | ltla ₁₇ |
| | 28+(i-1)x40 | ltla ₃₂ | ltla ₃₁ | ltla ₃₀ | ltla ₂₉ | ltla ₂₈ | ltla ₂₇ | ltla ₂₆ | ltla ₂₅ |
| LTP/FTP longitude (ltlo) | 29+(i-1)x40 | ltlo ₈ | ltlo ₇ | ltlo ₆ | ltlo ₅ | ltlo ₄ | ltlo ₃ | ltlo ₂ | ltlo ₁ |
| | 30+(i-1)x40 | ltlo ₁₆ | ltlo ₁₅ | ltlo ₁₄ | ltlo ₁₃ | ltlo ₁₂ | ltlo ₁₁ | ltlo ₁₀ | ltlo ₉ |
| | 31+(i-1)x40 | ltlo ₂₄ | ltlo ₂₃ | ltlo ₂₂ | ltlo ₂₁ | ltlo ₂₀ | ltlo ₁₉ | ltlo ₁₈ | ltlo ₁₇ |
| | 32+(i-1)x40 | ltlo ₃₂ | ltlo ₃₁ | ltlo ₃₀ | ltlo ₂₉ | ltlo ₂₈ | ltlo ₂₇ | ltlo ₂₆ | ltlo ₂₅ |
| LTP/FTP height (lth) | 33+(i-1)x40 | lth ₈ | lth ₇ | lth ₆ | lth ₅ | lth ₄ | lth ₃ | lth ₂ | lth ₁ |
| | 34+(i-1)x40 | lth ₁₆ | lth ₁₅ | lth ₁₄ | lth ₁₃ | lth ₁₂ | lth ₁₁ | lth ₁₀ | lth ₉ |
| DFPAP latitude (fpla) | 35+(i-1)x40 | fpla ₈ | fpla ₇ | fpla ₆ | fpla ₅ | fpla ₄ | fpla ₃ | fpla ₂ | fpla ₁ |
| | 36+(i-1)x40 | fpla ₁₆ | fpla ₁₅ | fpla ₁₄ | fpla ₁₃ | fpla ₁₂ | fpla ₁₁ | fpla ₁₀ | fpla ₉ |
| | 37+(i-1)x40 | fpla ₂₄ | fpla ₂₃ | fpla ₂₂ | fpla ₂₁ | fpla ₂₀ | fpla ₁₉ | fpla ₁₈ | fpla ₁₇ |
| DFPAP longitude (fplo) | 38+(i-1)x40 | fplo ₈ | fplo ₇ | fplo ₆ | fplo ₅ | fplo ₄ | fplo ₃ | fplo ₂ | fplo ₁ |
| | 39+(i-1)x40 | fplo ₁₆ | fplo ₁₅ | fplo ₁₄ | fplo ₁₃ | fplo ₁₂ | fplo ₁₁ | fplo ₁₀ | fplo ₉ |
| | 40+(i-1)x40 | fplo ₂₄ | fplo ₂₃ | fplo ₂₂ | fplo ₂₁ | fplo ₂₀ | fplo ₁₉ | fplo ₁₈ | fplo ₁₇ |
| approach threshold crossing height (apth) | 41+(i-1)x40 | apth ₈ | apth ₇ | apth ₆ | apth ₅ | apth ₄ | apth ₃ | apth ₂ | apth ₁ |
| approach TCH units selector (aptu) | 42+(i-1)x40 | aptu ₁ | apth ₁₅ | apth ₁₄ | apth ₁₃ | apth ₁₂ | apth ₁₁ | apth ₁₀ | apth ₉ |
| glide path angle (gpa) | 43+(i-1)x40 | gpa ₈ | gpa ₇ | gpa ₆ | gpa ₅ | gpa ₄ | gpa ₃ | gpa ₂ | gpa ₁ |
| | 44+(i-1)x40 | gpa ₁₆ | gpa ₁₅ | gpa ₁₄ | gpa ₁₃ | gpa ₁₂ | gpa ₁₁ | gpa ₁₀ | gpa ₉ |
| course width (cw) | 45+(i-1)x40 | cw ₈ | cw ₇ | cw ₆ | cw ₅ | cw ₄ | cw ₃ | cw ₂ | cw ₁ |
| DLength offset (dlo) | 46+(i-1)x40 | dlo ₈ | dlo ₇ | dlo ₆ | dlo ₅ | dlo ₄ | dlo ₃ | dlo ₂ | dlo ₁ |
| FAS CRC (fc) | 47+(i-1)x40 | fc ₈ | fc ₇ | fc ₆ | fc ₅ | fc ₄ | fc ₃ | fc ₂ | fc ₁ |
| | 48+(i-1)x40 | fc ₁₆ | fc ₁₅ | fc ₁₄ | fc ₁₃ | fc ₁₂ | fc ₁₁ | fc ₁₀ | fc ₉ |
| | 49+(i-1)x40 | fc ₂₄ | fc ₂₃ | fc ₂₂ | fc ₂₁ | fc ₂₀ | fc ₁₉ | fc ₁₈ | fc ₁₇ |
| | 50+(i-1)x40 | fc ₃₂ | fc ₃₁ | fc ₃₀ | fc ₂₉ | fc ₂₈ | fc ₂₇ | fc ₂₆ | fc ₂₅ |
| FAS vertical alert limit/approach status (va) | 51+(i-1)x40 | va ₈ | va ₇ | va ₆ | va ₅ | va ₄ | va ₃ | va ₂ | va ₁ |
| FAS lateral alert limit/approach status (la) | 52+(i-1)x40 | la ₈ | la ₇ | la ₆ | la ₅ | la ₄ | la ₃ | la ₂ | la ₁ |
| GNS-B message CRC (gc) | 53+(i-1)x40 | gc ₈ | gc ₇ | gc ₆ | gc ₅ | gc ₄ | gc ₃ | gc ₂ | gc ₁ |
| | 54+(i-1)x40 | gc ₁₆ | gc ₁₅ | gc ₁₄ | gc ₁₃ | gc ₁₂ | gc ₁₁ | gc ₁₀ | gc ₉ |
| | 55+(i-1)x40 | gc ₂₄ | gc ₂₃ | gc ₂₂ | gc ₂₁ | gc ₂₀ | gc ₁₉ | gc ₁₈ | gc ₁₇ |
| | 56+(i-1)x40 | gc ₃₂ | gc ₃₁ | gc ₃₀ | gc ₂₉ | gc ₂₈ | gc ₂₇ | gc ₂₆ | gc ₂₅ |
| response reservation field | 57+(i-1)x40 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| c | 58+(i-1)x40 | c ₁₆ | c ₁₅ | c ₁₄ | c ₁₃ | c ₁₂ | c ₁₁ | c ₁₀ | c ₉ |
| c | 59+(i-1)x40 | c ₈ | c ₇ | c ₆ | c ₅ | c ₄ | c ₃ | c ₂ | c ₁ |
| flag | - | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

7.4.3.2 Test cases

The equipment under test must 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 must 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.

Whenever a burst is specified in a test without values being given for all the parameters in a burst, and where the test does not instantiate the values, then the values of these parameters may be ignored.

If an expected test result mentioned in a test step is not observed during the execution of a test case, then the test case must 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 EN 302 842-2 [5] clause 7.4.3.2.1.

| Macro Name: M_SYNC_BURST_SEND (slots, range) | | Sends sync burst transmissions from other simulated stations over a number of slots. | | | | |
|---|-------------|--|------------|---|------------|---|
| Parameters: (slots = number of slots to transmit over; range = range of the simulated stations) | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| macro | 1 | record | | p = truncate(slots / 32) q = slots - p × 32 r = truncate(q / 16) s = q - r × 16 | | Set p to the number of times the number of slots divides by 32. Set q to the number of remainder slots after division by 32. Set r to the number of times the remainder divides by 16 (0 or 1). Set s to the further remainder after q is divided by 16. |
| | 2 | rep p | | | | Repeat over p slots. |
| | 3 | send | RF | SYNC_BURST_k(16) (s:= add_X; lat:= CPR_LAT(0); lon:= CPR_LON(E range NM); pt:= 3; po:= 0) | Sk(16) | Send a 16-slot sync burst from simulated station X which is <range> NM from the station under test. |
| | 4 | send | RF | SYNC_BURST_k(16) (s:= add_Y; lat:= CPR_LAT(0); lon:= CPR_LON(W range NM); pt:= 3; po:= 0) | Sk(16) | Send a 16-slot sync burst from simulated station Y which is <range> NM from the station under test. |
| | 5 | endrep | | | | End loop. |
| | 6 | rep r | | | | Repeat over p slots. |
| | 7 | send | RF | SYNC_BURST_k(16) (s:= add_X; lat:= CPR_LAT(0); lon:= CPR_LON(E range NM); pt:= 3; po:= 0) | Sk(16) | Send a 16-slot sync burst from simulated station X which is <range> NM from the station under test. |
| | 8 | endrep | | | | End loop. |
| | 9 | send | RF | SYNC_BURST_k(s) (s:= add_Z; lat:= CPR_LAT(N range NM); lon:= CPR_LON(0); pt:= 3; po:= 0) | Sk(16) | Send a sync burst that is s slots long from simulated station Z which is <range> NM from the station under test. |

Comments: Sends sync bursts to the station under test which can be used as a means to reserve a large group of slots in the following superframe.

7.4.3.2.2 Test case descriptions

| Test Case Name: ADSB_Request_A | | | | | | |
|---|------|--------|-------|---|-----|--|
| Purpose: To demonstrate that a station which desires another station to transmit a single autonomous synchronization burst will transmit an ADS-B request burst with the sleep and auto bits set to zero. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | SYNC_BURST_b (s:= add_B) | Sb | Send a sync burst from station B. |
| | 3 | send | Appln | REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (SINGLE RESPONSE, AUTONOMOUS SELECTION) to station B | | Instruct the station under test to send an ADS-B request to station B, requesting a single response in the specified slot and autonomous selection of variable field by station B. |
| | 4 | await | RF | ADSB_REQUEST_a (s = add_A, d = add_B) transmitted by station A | ARa | Wait for the ADS-B request to be transmitted by station A. |
| | 5 | verify | RF | For ADSB_REQUEST_a (s = add_A; d = add_B): burst is a general request with r-mi ₁ := 0; includes unicast reservation field; has burst format as defined in clause 7.4.3.1.5; sleep = 0; auto = 0 | | Verify that a general request burst has been sent by station A to station B with the format in clause 7.4.3.1.5 and with sleep = 0 and auto = 0. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Request_B | | | | | | |
|---|------|--------|-------|--|-----|---|
| Purpose: To demonstrate that a station which desires another station to transmit a single autonomous synchronization burst with a specified variable part will transmit an ADS-B request burst with sleep=0 and auto=1 and include the auto parameters. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | SYNC_BURST_b (s:= add_B) | Sb | Send a sync burst from station B. |
| | 3 | send | Appln | REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (SINGLE RESPONSE, BASIC VARIABLE PART, GEOMETRIC ALTITUDE RATE) TO STATION B | | Instruct station A to send an ADS-B request to station B, requesting a single response, a specified variable part in the specified slot, and a specified type of altitude rate. |
| | 4 | await | RF | ADSB_REQUEST_b (s = add_A; d = add_B) sent by station A | ARb | Wait for the ADS-B request to be transmitted by station A. |

| Test Case Name: ADSB_Request_B | | | | | | |
|---|------|--------|-----|--|-----|--|
| Purpose: To demonstrate that a station which desires another station to transmit a single autonomous synchronization burst with a specified variable part will transmit an ADS-B request burst with sleep=0 and auto=1 and include the auto parameters. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 5 | verify | RF | For ADSB_REQUEST_b (s = add_A, d = add_B): burst is a general request with r-mi ₁ = 0; includes unicast reservation field; has burst format as defined in clause 7.4.3.1.5; sleep = 0; auto = 1; r-id = 0; aux = 1 | | Verify that a general request burst has been sent by station A to station B with the format in clause 7.4.3.1.5 and with sleep = 0 and auto = 1. Verify that the auto parameters are included with r-id = 0 and aux = 1. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Request_C | | | | | | |
|---|------|--------|-------|---|-----|---|
| Purpose: To demonstrate that a station which desires another station to transmit synchronization bursts autonomously with the use of sleep mode, will transmit an ADS-B request burst with sleep=1 and auto=0 and include the sleep parameters. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | SYNC_BURST_b (s:= add_B) | Sb | Send a sync burst from station B. |
| | 3 | send | Appln | REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (SLEEP MODE, AUTONOMOUS SELECTION) (snr:=15; vel:=10; pos:=100) TO STATION B | ARc | Instruct station A to send an ADS-B request to station B, requesting sleep mode and autonomous selection of variable field by station B. |
| | 4 | await | RF | ADSB_REQUEST_c (s = add_A; d = add_B) sent by station A | ARc | Wait for the ADS-B request to be transmitted by station A. |
| | 5 | verify | RF | For ADSB_REQUEST_c (s:= add_A; d:= add_B): burst is a general request with r-mi ₁ = 0; includes autotune reservation field; has the burst format defined in clause 7.4.3.1.5; sleep:= 1; auto = 0; snr = 15; vel = 10; pos = 100 | | Verify that a general request burst has been sent by station A to station B with the format in clause 7.4.3.1.5, with sleep = 1 and auto = 0, the sleep parameters included, and the correct values of snr, vel and pos used. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | ADSB_Request_D | | | | |
|------------------|------|---|-------|--|-----|--|
| Purpose: | | To demonstrate that a station which desires another station to transmit synchronization bursts with a specified variable part and with the use of sleep mode, will transmit an ADS-B request burst with sleep=1 and auto=1 and include the auto and sleep parameters. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | SYNC_BURST_b (s:= add_B) | Sb | Send a sync burst from station B. |
| | 3 | send | Appln | REQUEST TO TRANSMIT ADSB REQUEST MESSAGE (SLEEP MODE, FULL POSITION VARIABLE PART) (snr:=6, vel:= 5, pos:= 150) TO STATION B | | Instruct station A to send an ADS-B request to station B, requesting use of sleep mode and a specified variable part in the specified slots. |
| | 4 | await | RF | ADSB_REQUEST_d (s = add_A, d = add_B) sent by station A | ARd | Wait for the ADS-B request to be transmitted by station A. |
| | 5 | verify | RF | For ADSB_REQUEST_d (s = add_A, d =add_B): burst is a general request with r-mi ₁ = 0; includes autotune reservation field; has the burst format defined in clause 7.4.3.1.5; sleep = 1; auto = 1; snr = 15; vel = 10; pos = 100 | ARd | Verify that a general request burst has been sent by station A to station B with the format in clause 7.4.3.1.5, with sleep = 1 and auto = 1, the sleep and auto parameters included and the correct values of snr, vel and pos. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Reporting | | | | | | |
|---|------|--------|-----------------------------|---|-----|---|
| Purpose: To demonstrate that a station will send at least 12 sync bursts a minute averaged over the two GSCs. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 60 | | n:= 1 | | Set n equal to 1. |
| | 3 | await | RF (GSC1 and GSC2) | SYNC_BURST_b (s = add_A) | Sb | Monitoring both GSC1 and GSC2, await a sync burst from station A. |
| | 4 | record | RF (GSC1 and GSC2) | time(n):= time at the beginning of slot containing SYNC_BURST_b (s = add_A) | Sb | Record the time at which each sync burst is transmitted. |
| | 5 | verify | | time(n) - time(n - 1) = 5 ± 2 s | | Verify that the bursts are transmitted about 5 seconds apart. |
| | 6 | endrep | | n:= n + 1 | | Add 1 to the value of n and repeat test on GSC1/GSC2 60 times. |
| | 7 | verify | | time(60) - time(1) ≤ 5 mins | | Verify that averaged over the two GSCs the station emits on average at least 12 sync bursts per minute. |
| postamble | 8 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Sleep_A | | | | | | |
|--|------|--------|----------|--|-----|--|
| Purpose: To demonstrate that the sleep mode parameters are obeyed according to the position and velocity of the station under test, and that the sleep mode is overridden by a directed request. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 3 | | p:= {125, 0, 175} v:= {0, 12, 12} | | Repeat for different values of position and velocity. |
| | 3 | send | RF | ADSB_Request_c (or:= 0; dt:= 14; do:= 0; lg:= 0; rcvr:= 11; f:= f1; trmt:= 1; s:= add_B; d:= add_A; sleep:= 1; auto:= 0; snr:= 15; vel:= 10; pos:= 100; nr:= 12) | ARc | Send a directed request with an ADSB request with sleep bit=1 and non-zero values for pos and vel to station A. |
| | 4 | send | Position | Input position POS:= 0 | | Input position of station A. |
| | 5 | send | Velocity | Input velocity VEL:= 0 | | Input velocity of station A. |
| | 6 | rep 60 | | n:= 1 | | Start loop. Set n to equal 1. |
| | 7 | await | RF | SYNC_BURST_b (s = add_A) | Sb | Wait for a sync burst from station A. |
| | 8 | record | RF | time(n):= time at beginning of slot containing SYNC_BURST_b (s = add_A) | Sb | Record the time at which each sync burst is transmitted. |
| | 9 | endrep | | n:= n + 1 | | End loop. |
| | 10 | verify | RF | time(60) - time(1) = 5 mins | | Confirm that station A transmits according to the ADSB request. The directed request specifies nr = 12, so 12 transmissions per minute are expected. |
| | 11 | send | Position | Input current position as POS:= p | | Input a new position. |

| Test Case Name: ADSB_Sleep_A | | | | | | |
|--|------|--------|----------|--|-----|---|
| Purpose: To demonstrate that the sleep mode parameters are obeyed according to the position and velocity of the station under test, and that the sleep mode is overridden by a directed request. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 12 | send | Velocity | Input current velocity as VEL:= v | | Input a new velocity. |
| | 13 | wait | | 60 seconds | | Allow time for the station A to set up new streams using the incremental broadcast procedures. |
| | 14 | rep 60 | | n:= 1 | | Start loop. Repeat test 60 times. |
| | 15 | await | RF | SYNC_BURST_b (s = add_A) | Sb | Wait for a sync burst from station A. |
| | 16 | record | RF | time(n): = time at the beginning of slot containing SYNC_BURST_b (s = add_A) | Sb | Record the time at which each sync burst is transmitted. |
| | 17 | endrep | | n:= n + 1 | | End loop. |
| | 18 | verify | | time(60) - time(1) = 4 mins | | Verify that the station emits 15 sync bursts a minute as specified in the sleep parameters. |
| | 19 | send | RF | DIR_REQ_a (or:= 1; dt:= 5; nr:= 4; do:= 100; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; s:= add_B; d:= add_A) | Da | Send a directed request reservation from station B to station A with or = 1, causing previously placed directed request reservations to be cancelled. |
| | 20 | record | RF | dir_time:= time at beginning of slot containing DIR_REQ_a (s = add_B) | Da | Define a reference time to measure relative times from during the test. |
| | 21 | rep 20 | | n:= 1 | | Start loop. Repeat 20 times. |
| | 22 | await | RF | DIR_SYNC_BURST_a (s = add_A) | DSa | Wait for the next sync burst from station A. |
| | 23 | verify | RF | DIR_SYNC_BURST_a (s = add_A) contained in slot beginning at: time = dir_time + 100 × (60/M1) + (n - 1) × 15 | DSa | Verify that station A transmits according to the new directed request. |
| | 24 | endrep | | n:= n + 1 | | End loop. |
| | 25 | endrep | | next p, v | | End loop. |
| postamble | | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Sleep_B | | | | | | |
|--|------|--------|----------|---|--------------------------|--|
| Purpose: To demonstrate that the position and velocity sleep mode parameters are ignored when either pos or vel, respectively, are zero. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | ADSB_Request_c (or:= 0; dt:= 14; do:= 0; lg:= 0; rcvr:= 11; f:= f1; trmt:= 1; s:= add_B; d:= add_A; sleep:= 1; auto:= 0; snr:= 15; vel:= 10; pos:= 0; nr:= 8) | ARc | Send a directed request with an ADSB request with sleep bit=1, a non-zero value for vel, and pos=0 to station A. |
| | 3 | send | Position | Input current position as POS:= 0 | | Input position of station A as zero. |
| | 4 | send | Velocity | Input current velocity as VEL:= 0 | | Input velocity of station A as zero. |
| | 5 | rep 40 | | n:= 1 | | Start a loop. Set n equal to 1. |
| | 6 | await | RF | | SYNC_BURST_b (s = add_A) | Sb |

| Test Case Name: | | ADSB_Sleep_B | | | | |
|------------------|------|---|----------|--|-----|---|
| Purpose: | | To demonstrate that the position and velocity sleep mode parameters are ignored when either pos or vel, respectively, are zero. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 7 | record | RF | time(n):= time at the beginning of slot containing SYNC_BURST_b (s = add_A) | Sb | Record the time at which the sync burst is transmitted. |
| | 8 | end rep | | n:= n + 1 | | End loop. |
| | 9 | verify | | time(40) - time(1) = 5 mins | | Confirm that station A transmits according to the ADSB request. The directed request specifies nr= 8, therefore 8 transmissions per min are expected. |
| | 10 | send | Position | Input current position as POS:= 120 | | Input a new position that is significantly different from the first. |
| | 11 | rep 40 | | n:= 1 | | Start loop. Set n equal to 1. |
| | 12 | await | RF | SYNC_BURST_b (s = add_A) | Sb | Wait for a sync burst from station A. |
| | 13 | record | RF | time(n):= time at the beginning of slot containing SYNC_BURST_b (s = add_A) | Sb | Record the time at which the sync burst is transmitted. |
| | 14 | end rep | | n:= n + 1 | | End loop. |
| | 15 | verify | | time(40) - time(1) = 5 mins | | Confirm that the station continues to transmit according to the ADSB request. The ADSB request specifies nr= 8, therefore 8 transmissions per min are expected. |
| | 16 | send | RF | ADSB_Request_c (or:= 0; dt:= 14; do:= 0; lg:= 0; rcvr:= 11; f:= f1; trmt:= 1; s:= add_B; d:= add_A; sleep:= 1; auto:= 0; snr:= 15; vel:= 0; pos:= 100; nr:= 8) | ARc | Send a directed request with an ADSB request with sleep bit=1, a non-zero value for pos, and vel=0 to station A. |
| | 17 | send | Position | Input current position as POS:= 0 | | Input position of station A as zero. |
| | 18 | send | Velocity | Input current velocity as VEL:= 0 | | Input velocity of station A as zero. |
| | 19 | rep 40 | | n:= 1 | | Start a loop. Set n equal to 1. |
| | 20 | await | RF | SYNC_BURST_b (s = add_A) | Sb | Wait for a sync burst from station A. |
| | 21 | record | RF | time(n):= time at the beginning of slot containing SYNC_BURST_b (s = add_A) | Sb | Record the time at which the sync burst is transmitted. |
| | 22 | end rep | | n:= n + 1 | | End loop. |
| | 23 | verify | | time(40) - time(1) = 5 mins | | Confirm that station A transmits according to the ADSB request. The directed request specifies nr= 8, therefore 8 transmissions per min are expected. |
| | 24 | send | Velocity | Input current velocity as VEL:= 15 | | Input a new velocity that significantly exceeds the first velocity value. |
| | 25 | rep 40 | | n:= 1 | | Start loop. Set n equal to 1. |
| | 26 | await | RF | SYNC_BURST_b (s = add_A) | Sb | Wait for a sync burst from station A. |
| | 27 | record | RF | time(n):= time at the beginning of slot containing SYNC_BURST_b (s = add_A) | Sb | Record the time at which the sync burst is transmitted. |
| | 28 | end rep | | n:= n + 1 | | End loop. |
| | 29 | verify | | time(40) - time(1) = 5 mins | | Confirm that the station continues to transmit according to the ADSB request. The ADSB request specifies nr= 8, therefore 8 transmissions per min are expected. |
| postamble | 30 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | ADSB_Request_Time | | | | |
|------------------|--------|--|-----------|---|---------------------------------------|--|
| Purpose: | | To demonstrate that a station requesting the time synchronization information field (information field ID 2) will only transmit a request to a station that has announced that it is operating with a primary time source (tfom = 0 or 1). | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | SYNC_BURST_b (s:= add_B; tfom:= 2) | Sb | Sync burst from station B, secondary time given. |
| | 3 | send | RF | SYNC_BURST_b (s:= add_C; tfom:= 0) | Sb | Sync burst from station C, certified primary time given. |
| | 4 | send | RF | SYNC_BURST_b (s:= add_D; tfom:= 2) | Sb | Sync burst from station D, secondary time given. |
| | 5 | send | RF | SYNC_BURST_b (s:= add_E; tfom:= 1) | Sb | Sync burst from station E, non-certified primary time given. |
| | 6 | send | RF | SYNC_BURST_b (s:= add_F; tfom:= 0) | Sb | Sync burst from station F, certified primary time given. |
| | 7 | send | Appln | REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (s:=add_A; d:=add_B; r-id:=4) | | Instruct station A via the Appln PCO to send a request burst to station B requesting UTC time information field. |
| | 8 | send | Appln | REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (s:=add_A; d:=add_C; r-id:=4) | | Instruct station A via the Appln PCO to send a request burst to station C requesting UTC time information field. |
| | 9 | send | Appln | REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (s:=add_A; d:=add_D; r-id:=4) | | Instruct station A via the Appln PCO to send a request burst to station D requesting UTC time information field. |
| | 10 | send | Appln | REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (s:=add_A; d:=add_E; r-id:=4) | | Instruct station A via the Appln PCO to send a request burst to station E requesting UTC time information field. |
| | 11 | send | Appln | REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (s:=add_A; d:=add_F; r-id:=4) | | Instruct station A via the Appln PCO to send a request burst to station F requesting UTC time information field. |
| | 12 | rep 5 | | n:= 1 | | Start loop. |
| | 13 | await | RF | ADSB_REQUEST_b (s = add_A) | ARb | Wait for an ADSB request message. |
| | 14 | record | RF | DEST(n) = value of d given in ADSB_REQUEST_b (s = add_A) | ARb | Record the destination address in the ADSB request. |
| | 15 | verify | RF | DEST(n) = add_C, add_E or add_F not add_B or add_D | | Verify that requests were only sent to those that had declared primary time. |
| 16 | endrep | | n:= n + 1 | | End loop. | |
| postamble | 17 | | | | Bring test equipment into idle state. | |
| Comments: | | | | | | |

| Test Case Name: ADSB_Priority | | | | | | |
|---|------|---------|-------|--|-----|---|
| Purpose: To demonstrate that information fields 0, 1, 2, 3, 4 and A1 hex have priority over other information fields. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | rep 10 | | n:= 1 | | Set n equal to 1. |
| test body | 3 | send | Appln | REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (SINGLE RESPONSE, AUTONOMOUS SELECTION) TO STATION B | | Instruct station A to send an ADS-B request to station B, requesting a single response in the specified slot and autonomous selection of the variable field by station B. |
| | 4 | await | RF | ADSB_REQUEST_a (s:= add_A; d: = add_B) sent by station A | ARa | Wait for the ADS-B request to be transmitted by station A. |
| | 5 | verify | | r-id is equal to one of {0, 1, 2, 3, 4, A1} hex | | Show that station A chooses an information field from the set {0, 1, 2, 3, 4, A1}. |
| | 6 | end rep | | n:= n + 1 | | End loop. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_First_TCP_A | | | | | | |
|---|------|--------|-------|---|-----|---|
| Purpose: To demonstrate that when the status of its first TCP changes, a mobile station will autonomously transmit a series of synchronization bursts containing the single-slot TCP variable part. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | Appln | TCP and SVQ data | | Send TCP data to station A that may be transmitted. |
| | 3 | rep 3 | | n:= 1 | | Start loop. |
| | 4 | await | RF | SYNC_BURST_c (s = add_A) | Sc | Wait for an autonomous sync burst from station A. |
| | 5 | verify | RF | tqc = 1 in SYNC_BURST_c (s = add_A) | Sc | Verify that the TCP/SVQ change flag is set to 1. |
| | 6 | endrep | | | | End loop. |
| | 7 | send | Appln | New TCP data indicating change to first TCP | | Send new TCP data to station A indicating a change to its first TCP. |
| | 8 | record | | data_time:= time that new TCP data was sent | | Record the time that the new TCP data was sent to the station under test. |
| | 9 | rep 6 | | n:= 1 | | Start loop. |
| | 10 | verify | RF | SYNC_SINGLE_SLOT_TCP_a (s = add_A) transmitted before time = data_time + 2 × n with a/d = 0 | SSa | Verify that station A transmits autonomously a sync burst with the single slot TCP variable part. |
| | 11 | endrep | | n:= n + 1 | | End loop. |
| postamble | 12 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Status_SVQ_A | | | | | | |
|--|------|--------|-------|--|-----|---|
| Purpose: To demonstrate that when the status of its SVQ changes, a mobile station will autonomously transmit a series of synchronization bursts containing the single-slot SVQ variable part. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | Appln | TCP and SVQ data | | Send TCP data to station A that may be transmitted. |
| | 3 | rep 3 | | n:= 1 | | Start loop. |
| | 4 | await | RF | SYNC_BURST_c (s = add_A) | Sc | Wait for an autonomous sync burst from station A. |
| | 5 | verify | RF | tqc = 1 in SYNC_BURST_c (s = add_A) | Sc | Verify that the TCP/SVQ change flag is set to 1. |
| | 6 | endrep | | | | End loop. |
| | 7 | send | Appln | New SVQ data indicating change to NAC _v | | Send new TCP data to station A indicating a change to NAC _v . |
| | 8 | record | | data_time:= time that new TCP data was sent | | Record the time that the new TCP data was sent to the station under test. |
| | 9 | rep 6 | | n:= 1 | | Start loop. |
| | 10 | verify | RF | SYNC_SINGLE_SLOT_SVQ_a (s = add_A) transmitted before time = data_time + 2 × n with a/d = 0 | SVa | Verify that station A transmits autonomously a sync burst with the single slot SVQ variable part. |
| | 11 | endrep | | n:= n + 1 | | End loop. |
| postamble | 12 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_TCP/SVQ_Flag_A | | | | | | |
|--|------|--------|-------|--|-------|--|
| Purpose: To demonstrate that a station which is transmitting a series of directed synchronization bursts that include the two-slot TCP/SVQ variable part will indicate when there is a change to any of its TCPs or to its SVQ indicators by setting the TCP/SVQ change flag to zero in all directed synchronization bursts transmitted by the station. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | Appln | TCP and SVQ data | | Send TCP data to station A that may be transmitted. |
| | 3 | send | RF | ADSB_Request_e (s:= add_B; d:= add_A; sleep:= 0; auto:= 1; r-b/a:= 0; lg:= 0; aux:= 3; r- id:= 2 hex; dt:= 14; f:= f1; lg:= 1; trmt:= 1; do:= 0; or:= 0; rcvr:= 11; nr:= 8) | ARe | Send an ADS-B request with a directed request to the station under test, requesting the station to transmit sync bursts with the full position variable field. |
| | 4 | send | RF | ADSB_Request_e (s:= add_B; d:= add_A; sleep:= 0; auto:= 1; r-b/a:= 0; lg:= 0; aux:= 3; r-id:= 0 hex; dt:= 14; f:= f1; lg:= 0; trmt:= 1; do:= 0; or:= 0; rcvr:= 11; nr:= 8) | ARe | Send an ADS-B request with a directed request to the station under test, requesting the station to transmit sync bursts with the basic variable field. |
| | 5 | send | RF | ADSB_Request_e (s:= add_B; d:= add_A; sleep:= 0; auto:= 1; r-b/a:= 0; lg:= 0; aux:= 3; r-id:= 8 hex; dt:= 14; f:= f1; lg:= 1; trmt:= 1; do:= 0; or:= 0; rcvr:= 11; nr:= 8) | ARe | Send an ADS-B request with a directed request to the station under test, requesting the station to transmit sync bursts with the two-slot TCP variable field. |
| | 6 | rep 3 | | | n:= 1 | |

| Test Case Name: | | ADSB_TCP/SVQ_Flag_A | | | | |
|-----------------|------|--|-------|--|-----|---|
| Purpose: | | To demonstrate that a station which is transmitting a series of directed synchronization bursts that include the two-slot TCP/SVQ variable part will indicate when there is a change to any of its TCPs or to its SVQ indicators by setting the TCP/SVQ change flag to zero in all directed synchronization bursts transmitted by the station. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 7 | await | RF | DIR_SYNC_BURST_b (s = add_A) | DSb | Wait for a directed sync burst with full position variable part from station A. |
| | 8 | verify | RF | tqc = 1 in DIR_SYNC_BURST_b (s = add_A) | DSb | Verify that the TCP/SVQ change flag is set to 1. |
| | 9 | await | RF | DIR_SYNC_BURST_a (s = add_A) | DSa | Wait for a directed sync burst with basic variable part from station A. |
| | 10 | verify | RF | tqc = 1 in DIR_SYNC_BURST_a (s = add_A) | DSa | Verify that the TCP/SVQ change flag is set to 1. |
| | 11 | await | RF | DIR_SYNC_BURST_c (s = add_A) | DSc | Wait for a directed sync burst with two-slot TCP variable part from station A. |
| | 12 | verify | RF | tqc = 1 in DIR_SYNC_BURST_c (s = add_A) | DSc | Verify that the TCP/SVQ change flag is set to 1. |
| | 13 | endrep | | | | End loop. |
| | 14 | send | Appln | New TCP data indicating change to TCP+3 | | Send new TCP data to station A indicating a change to TCP+3. |
| | 15 | await | RF | DIR_SYNC_BURST_b (s = add_A) | DSb | Wait for the next directed sync burst with full position variable part from station A. |
| | 16 | verify | RF | tqc = 0 in DIR_SYNC_BURST_b (s = add_A) | DSb | Verify that the TCP/SVQ change flag is set to 0. |
| | 17 | await | RF | DIR_SYNC_BURST_a (s = add_A) | DSa | Wait for the next directed sync burst with basic variable part from station A. |
| | 18 | verify | RF | tqc = 0 in DIR_SYNC_BURST_a (s = add_A) | DSa | Verify that the TCP/SVQ change flag is set to 0. |
| | 19 | await | RF | DIR_SYNC_BURST_c (s = add_A) | DSc | Wait for the next directed sync burst with two-slot TCP/SVQ variable part from station A. |
| | 20 | verify | RF | DIR_SYNC_BURST_c (s = add_A) contains: new TCP data; tqc = 1 | DSc | Verify that the TCP/SVQ change flag is set to 1. |
| | 21 | rep 3 | | n:= 1 | | Start loop. |
| | 22 | await | RF | DIR_SYNC_BURST_b (s = add_A) | DSb | Wait for a directed sync burst with full position variable part from station A. |
| | 23 | verify | RF | tqc = 1 in DIR_SYNC_BURST_b (s = add_A) | DSb | Verify that the TCP/SVQ change flag is set to 1. |
| | 24 | await | RF | DIR_SYNC_BURST_a (s = add_A) | DSa | Wait for a directed sync burst with basic variable part from station A. |
| | 25 | verify | RF | tqc = 1 in DIR_SYNC_BURST_a (s = add_A) | DSa | Verify that the TCP/SVQ change flag is set to 1. |
| | 26 | await | RF | DIR_SYNC_BURST_c (s = add_A) | DSc | Wait for a directed sync burst with two-slot TCP/SVQ variable part from station A. |
| | 27 | verify | RF | tqc = 1 in DIR_SYNC_BURST_c (s = add_A) | DSc | Verify that the TCP/SVQ change flag is set to 1. |
| | 28 | endrep | | | | End loop. |
| | 29 | send | Appln | New SVQ data indicating change to SIL | | Send new SVQ data to station A indicating a change to SIL. |
| | 30 | await | RF | DIR_SYNC_BURST_b (s = add_A) | DSb | Wait for the next directed sync burst with full position variable part from station A. |
| | 31 | verify | RF | tqc = 0 in DIR_SYNC_BURST_b (s = add_A) | DSb | Verify that the TCP/SVQ change flag is set to 0. |
| | 32 | await | RF | DIR_SYNC_BURST_a (s = add_A) | DSa | Wait for the next directed sync burst with basic variable part from station A. |
| | 33 | verify | RF | tqc = 0 in DIR_SYNC_BURST_a (s = add_A) | DSa | Verify that the TCP/SVQ change flag is set to 0. |

| Test Case Name: ADSB_TCP/SVQ_Flag_A | | | | | | |
|--|------|--------|-----|--|-----|---|
| Purpose: To demonstrate that a station which is transmitting a series of directed synchronization bursts that include the two-slot TCP/SVQ variable part will indicate when there is a change to any of its TCPs or to its SVQ indicators by setting the TCP/SVQ change flag to zero in all directed synchronization bursts transmitted by the station. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 34 | await | RF | DIR_SYNC_BURST_c (s = add_A) | DSc | Wait for the next directed sync burst with two-slot TCP/SVQ variable part from station A. |
| | 35 | verify | RF | DIR_SYNC_BURST_c (s = add_A) contains: new SVQ data; tqc = 1 | DSc | Verify that the TCP/SVQ change flag is set to 1. |
| | 36 | rep 3 | | n:= 1 | | Start loop. |
| | 37 | await | RF | DIR_SYNC_BURST_b (s = add_A) | DSb | Wait for a directed sync burst with full position variable part from station A. |
| | 38 | verify | RF | tqc = 1 in DIR_SYNC_BURST_b (s = add_A) | DSb | Verify that the TCP/SVQ change flag is set to 1. |
| | 39 | await | RF | DIR_SYNC_BURST_a (s = add_A) | DSa | Wait for a directed sync burst with basic variable part from station A. |
| | 40 | verify | RF | tqc = 1 in DIR_SYNC_BURST_a (s = add_A) | DSa | Verify that the TCP/SVQ change flag is set to 1. |
| | 41 | await | RF | DIR_SYNC_BURST_c (s = add_A) | DSc | Wait for a directed sync burst with two-slot TCP/SVQ variable part from station A. |
| | 42 | verify | RF | tqc = 1 in DIR_SYNC_BURST_c (s = add_A) | DSc | Verify that the TCP/SVQ change flag is set to 1. |
| | 43 | endrep | | | | End loop. |
| postamble | 44 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_TCP/SVQ_Flag_B | | | | | | |
|---|------|--------|-----|--|-----|--|
| Purpose: To show that a station which is transmitting a series of directed synchronization bursts which do not include the two slot TCP/SVQ variable part will set the TCP/SVQ change flag to 1. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | ADSB_Request_e (s:= add_B; d:= add_A; sleep:= 0; auto:= 1; r-b/a:= 0; lg:= 0; aux:= 3; r-id:= 2 hex; dt:= 14; f:= f1; lg:= 1; trmt:= 1; do:= 0; or:= 0; rcvr:= 11; nr:= 8) | ARe | Send an ADS-B request with a directed request to the station under test, requesting the station to transmit sync bursts with the full position variable field. |
| | 3 | send | RF | ADSB_Request_e (s:= add_B; d:= add_A; sleep:= 0; auto:= 1; r-b/a:= 0; lg:= 0; aux:= 3; r-id:= 0 hex; dt:= 14; f:= f1; lg:= 0; trmt:= 1; do:= 0; or:= 0; rcvr:= 11; nr:= 8) | ARe | Send an ADS-B request with a directed request to the station under test, requesting the station to transmit sync bursts with the basic variable field. |
| | 4 | send | RF | ADSB_Request_e (s:= add_B; d:= add_A; sleep:= 0; auto:= 1; r-b/a:= 0; lg:= 0; aux:= 3; r-id:= 1 hex; dt:= 14; f:= f1; lg:= 1; trmt:= 1; do:= 0; or:= 0; rcvr:= 11; nr:= 8) | ARe | Send an ADS-B request with a directed request to the station under test, requesting the station to transmit sync bursts with the high dynamic variable field. |
| | 5 | rep 3 | | n:= 1 | | Start loop. |
| | 6 | await | RF | DIR_SYNC_BURST_b (s = add_A) | DSb | Wait for a directed sync burst with full position variable part from station A. |

| Test Case Name: AD SB_TCP/SVQ_Flag_B | | | | | | |
|--|------|--------|-----|---|-----|--|
| Purpose: To show that a station which is transmitting a series of directed synchronization bursts which do not include the two slot TCP/SVQ variable part will set the TCP/SVQ change flag to 1. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 7 | verify | RF | tqc = 1 in DIR_SYNC_BURST_b (s = add_A) | DSb | Verify that the TCP/SVQ change flag is set to 1. |
| | 8 | await | RF | DIR_SYNC_BURST_a (s = add_A) | DSa | Wait for a directed sync burst with basic variable part from station A. |
| | 9 | verify | RF | tqc = 1 in DIR_SYNC_BURST_a (s = add_A) | DSa | Verify that the TCP/SVQ change flag is set to 1. |
| | 10 | await | RF | DIR_SYNC_BURST_c (s = add_A) | DSc | Wait for a directed sync burst with high dynamic variable part from station A. |
| | 11 | verify | RF | tqc = 1 in DIR_SYNC_BURST_c (s = add_A) | DSc | Verify that the TCP/SVQ change flag is set to 1. |
| | 12 | endrep | | | | End loop. |
| postamble | 13 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: AD SB_Basic_A | | | | | | |
|---|------|--------|-------|---|-----|---|
| Purpose: To demonstrate that a station will broadcast a basic sync burst message with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:={P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | AppIn | REQUEST TO TRANSMIT BASIC SYNC BURST (ADSB_BASIC_APPIN_PARAMETERS(x)) | | Send an instruction at the AppIn PCO to the station under test to transmit a basic sync burst. |
| | 3 | await | RF | BURST (s = add_A) broadcast by station A | | Wait for the message to be broadcast by the station under test. |
| | 4 | verify | RF | BURST (s = add_A) has the format of "SYNC_BASIC_a" | SBa | Verify that the transmitted burst conforms to "SYNC_BASIC_a" as set out in the burst format clause. |
| | 5 | verify | RF | ADSB_BASIC_RF_OUT_PARAMETERS(x) | | Verify that the parameters have been correctly encoded in the message. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: AD SB_Basic_Rec_A | | | | | | |
|--|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received basic sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:={P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | RF | SYNC_BASIC_a (ADSB_BASIC_RF_IN_PARAMETERS(x)) | SBa | Send a basic sync burst to the station under test from a simulated station B. |
| | 3 | await | AppOut | SYNC_BASIC_a MESSAGE INFORMATION | SBa | Wait for the message information to be output by the station under test at the AppOut PCO. |
| | 4 | verify | AppOut | ADSB_BASIC_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by the station under test at the AppOut PCO is correct. |

| Test Case Name: ADSB_Basic_Rec_A | | | | | | |
|--|------|--------|-----|------------------|-----|---------------------------------------|
| Purpose: To demonstrate that a station will correctly process a received basic sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_High_Dynamic_A | | | | | | |
|--|------|--------|-------|---|-----|--|
| Purpose: To demonstrate that a station will broadcast a high dynamic sync burst with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:={P,Q,R,S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | AppIn | REQUEST TO TRANSMIT HIGH DYNAMIC SYNC BURST (ADSB_HIGH_DYNAMIC_APPIN_PARAMETERS(x)) | | Send an instruction at the AppIn PCO to the station under test to transmit a high dynamic sync burst. |
| | 3 | await | RF | BURST (s:= add_A) broadcast by station A | | Wait for the message to be broadcast by the station under test. |
| | 4 | verify | RF | BURST (s:= add_A) has the format of "SYNC_HIGH_DYNAMIC_a" | SHa | Verify that the transmitted burst conforms to "SYNC_HIGH_DYNAMIC_a" as set out in the burst format clause. |
| | 5 | verify | RF | ADSB_HIGH_DYNAMIC_RF_OUT_PARAMETER S(x) | | Verify that the parameters have been correctly encoded in the message. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_High_Dynamic_Rec_A | | | | | | |
|---|------|--------|--------|---|--------|---|
| Purpose: To demonstrate that a station will correctly process a received high dynamic sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x: {P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | RF | SYNC_HIGH_DYNAMIC_a (ADSB_HIGH_DYNAMIC_RF_IN_PARAMETERS(x)) | SHa | Send a high dynamic sync burst to the station under test from a simulated station B. |
| | 3 | await | AppOut | SYNC_HIGH_DYNAMIC_a MESSAGE INFORMATION | SHa | Wait for the message information to be output by the station under test at the AppOut PCO. |
| | 4 | verify | AppOut | ADSB_HIGH_DYNAMIC_APPOUT_PARAMETER S(x) | | Verify that the content of the information output by the station under test at the AppOut PCO is correct. |
| | | 5 | endrep | | next x | |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Full_Position_A | | | | | | |
|---|------|--------|-------|---|-----|---|
| Purpose: To demonstrate that a station will broadcast a full position sync burst with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:={P, Q, R, S} | | Repeat test for 4 sets of parameters |
| | 2 | send | AppIn | REQUEST TO TRANSMIT FULL POSITION SYNC BURST (ADSB_FULL_POSITION_APPIN_PARAMETERS(x)) | | Send an instruction at the AppIn PCO to the station under test to transmit a full position sync burst. |
| | 3 | await | RF | BURST (s:= add_A) broadcast by station A | | Wait for the message to be broadcast by the station under test. |
| | 4 | verify | RF | BURST (s:= add_A) has the format of "SYNC_FULL_POSITION_a" | SPa | Verify that the transmitted burst conforms to "SYNC_FULL_POSITION_a" as set out in the burst format clause. |
| | 5 | verify | RF | ADSB_FULL_POSITION_RF_OUT_PARAMETER S(x) | | Verify that the parameters have been correctly encoded in the message. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Full_Position_Rec_A | | | | | | |
|--|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received full position sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x: {P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | RF | SYNC_FULL_POSITION_a (ADSB_FULL_POSITION_RF_IN_PARAMETERS(x)) | SPa | Send a full position sync burst to the station under test from a simulated station B. |
| | 3 | await | AppOut | SYNC_FULL_POSITION_a MESSAGE INFORMATION | SPa | Wait for the message information to be output by the station under test at the AppOut PCO. |
| | 4 | verify | AppOut | ADSB_FULL_POSITION_APPOUT_PARAMETER S(x) | | Verify that the content of the information output by the station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Basic_Ground_Rec_A | | | | | | |
|---|------|--------|--------|--|-----|---|
| Purpose: To demonstrate that a station will correctly process a received basic ground sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | RF | SYNC_BASIC_GROUND_a (ADSB_BASIC_GROUND_RF_IN_PARAMETERS(x)) | BGa | Send a basic ground sync burst to the station under test from a simulated ground station G. |
| | 3 | await | AppOut | SYNC_BASIC_GROUND_a MESSAGE INFORMATION | BGa | Wait for the message information to be output by the station under test at the AppOut PCO. |
| | 4 | verify | AppOut | ADSB_BASIC_GROUND_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by the station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_UTC_Time_Rec_A | | | | | | |
|---|------|--------|--------|--|------|---|
| Purpose: To demonstrate that a station will correctly process a received UTC time sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | RF | SYNC_UTC_TIME_a (ADSB_UTC_TIME_RF_IN_PARAMETERS(x)) | UTCa | Send a UTC time sync burst to the station under test from a simulated ground station G. |
| | 3 | await | AppOut | SYNC_UTC_TIME_a MESSAGE INFORMATION | | Wait for the message information to be output by the station under test at the AppOut PCO. |
| | 4 | verify | AppOut | ADSB_UTC_TIME_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by the station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Two_Slot_TCP/SVQ_A | | | | | | |
|--|------|--------|-------|--|-----|---|
| Purpose: To demonstrate that a station will broadcast a two slot TCP/SVQ sync burst with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:={P,Q,R,S} | | Repeat test for 4 sets of parameters |
| | 2 | send | AppIn | REQUEST TO TRANSMIT A TWO SLOT TCP SYNC BURST (ADSB_TWO_SLOT_TCP/SVQ_APPIN_PARAMETERS(x)) | | Send an instruction at the AppIn PCO to the station under test to transmit a two-slot TCP/SVQ sync burst. |
| | 3 | await | RF | BURST (s:= add_A) broadcast by station A | | Wait for the message to be broadcast by the station under test. |

| Test Case Name: ADSB_Two_Slot_TCP/SVQ_A | | | | | | |
|--|------|--------|-----|---|-----|--|
| Purpose: To demonstrate that a station will broadcast a two slot TCP/SVQ sync burst with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 4 | verify | RF | BURST (s:= add_A) has the format of "SYNC_TWO_SLOT_TCP/SVQ_a" | STa | Verify that the transmitted burst conforms to "SYNC_TWO_SLOT_TCP/SVQ_a" as set out in the burst format clause. |
| | 5 | verify | RF | ADSB_TWO_SLOT_TCP/SVQ_RF_OUT_PARAM ETERS(x) | | Verify that the parameters have been correctly encoded in the message. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Two_Slot_TCP/SVQ_Rec_A | | | | | | |
|---|------|--------|--------|--|-----|---|
| Purpose: To demonstrate that a station will correctly process a received two slot TCP sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | RF | SYNC_TWO_SLOT_TCP/SVQ_a (ADSB_TWO_SLOT_TCP/SVQ_RF_IN_PARAM ETERS(x)) | STa | Send a two-slot TCP sync burst to the station under test from a simulated station B. |
| | 3 | await | AppOut | SYNC_TWO_SLOT_TCP/SVQ_a MESSAGE INFORMATION | STa | Wait for the message information to be output by the station under test at the AppOut PCO. |
| | 4 | verify | AppOut | ADSB_TWO_SLOT_TCP/SVQ_APPOUT_PARAM ETERS(x) | | Verify that the content of the information output by the station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Single_Slot_TCP_A | | | | | | |
|---|------|--------|-------|--|-----|---|
| Purpose: To demonstrate that a station will broadcast a single slot TCP sync burst with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:={P,Q,R,S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | AppIn | REQUEST TO TRANSMIT A SINGLE SLOT TCP BURST (ADSB_SINGLE_SLOT_TCP_APPIN_PARAMETERS(x)) | | Send an instruction at the AppIn PCO to the station under test to transmit a single-slot TCP sync burst. |
| | 3 | await | RF | BURST (s:= add_A) broadcast by station A | | Wait for the message to be broadcast by the station under test. |
| | 4 | verify | RF | BURST (s:= add_A) has the format of "SYNC_SINGLE_SLOT_TCP_a" | SSa | Verify that the transmitted burst conforms to "SYNC_SINGLE_SLOT_TCP_a" as set out in the burst format clause. |
| | 5 | verify | RF | ADSB_SINGLE_SLOT_TCP_RF_OUT_PARAMETERS(x) | | Verify that the parameters have been correctly encoded in the message. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Single_Slot_TCP_Rec_A | | | | | | |
|--|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received single slot TCP sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x: {P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | RF | SYNC_SINGLE_SLOT_TCP_a (ADSB_SINGLE_SLOT_TCP_RF_IN_PARAMETERS(x)) | SSa | Send a single slot TCP sync burst to the station under test from a simulated station B. |
| | 3 | await | AppOut | SYNC_SINGLE_SLOT_TCP_a MESSAGE INFORMATION | SSa | Wait for the message information to be output by the station under test at the AppOut PCO. |
| | 4 | verify | AppOut | ADSB_SINGLE_SLOT_TCP_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by the station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Single_Slot_SVQ_A | | | | | | |
|---|------|--------|-------|---|-----|---|
| Purpose: To demonstrate that a station will broadcast a single slot SVQ sync burst with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:={P,Q,R,S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | AppIn | REQUEST TO TRANSMIT A SINGLE SLOT SVQ SYNC BURST (ADSB_SINGLE_SLOT_SVQ_APPIN_PARAMETERS(x)) | | Send an instruction at the AppIn PCO to the station under test to transmit a single slot SVQ sync burst. |
| | 3 | await | RF | BURST (s:= add_A) broadcast by station A | | Wait for the message to be broadcast by the station under test. |
| | 4 | verify | RF | BURST (s:= add_A) has the format of "SYNC_SINGLE_SLOT_SVQ_a" | SVa | Verify that the transmitted burst conforms to "SYNC_SINGLE_SLOT_SVQ_a" as set out in the burst format clause. |
| | 5 | verify | RF | ADSB_SINGLE_SLOT_SVQ_RF_OUT_PARAMETERS(x) | | Verify that the parameters have been correctly encoded in the message. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Single_Slot_SVQ_Rec_A | | | | | | |
|--|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received single slot SVQ sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x: {P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | RF | SYNC_SINGLE_SLOT_SVQ_a (ADSB_SINGLE_SLOT_SVQ_RF_IN_PARAMETERS(x)) | SVa | Send a single-slot SVQ sync burst to the station under test from a simulated station B. |
| | 3 | await | AppOut | SYNC_SINGLE_SLOT_SVQ_a MESSAGE INFORMATION | SVa | Wait for the message information to be output by the station under test at the AppOut PCO. |
| | 4 | verify | AppOut | ADSB_SINGLE_SLOT_SVQ_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by the station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Aircraft_Data_A | | | | | | |
|--|------|--------|-------|---|-----|---|
| Purpose: To demonstrate that a station will broadcast an aircraft data sync burst with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:={P,Q,R,S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | AppIn | REQUEST TO TRANSMIT AN AIRCRAFT DATA SYNC BURST (ADSB_AIRCRAFT_DATA_APPIN_PARAMETER S(x)) | | Send an instruction at the AppIn PCO to the station under test to transmit an aircraft data sync burst. |
| | 3 | await | RF | BURST (s:= add_A) broadcast by station A | | Wait for the message to be broadcast by the station under test. |
| | 4 | verify | RF | BURST (s:= add_A) has the format of "SYNC_AIRCRAFT_DATA_a" | SDa | Verify that the transmitted burst conforms to "SYNC_AIRCRAFT_DATA_a" as set out in the burst format clause. |
| | 5 | verify | RF | ADSB_AIRCRAFT_DATA_RF_OUT_PARAMETER S(x) | | Verify that the parameters have been correctly encoded in the message. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_Aircraft_Data_Rec_A | | | | | | |
|--|------|--------|--------|--|-----|---|
| Purpose: To demonstrate that a station will correctly process a received aircraft data sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x: {P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | RF | SYNC_AIRCRAFT_DATA_a (ADSB_AIRCRAFT_DATA_RF_IN_PARAMETER S(x)) | SDa | Send an aircraft data sync burst to the station under test from a simulated station B. |
| | 3 | await | AppOut | SYNC_AIRCRAFT_DATA_a MESSAGE INFORMATION | SDa | Wait for the message information to be output by the station under test at the AppOut PCO. |
| | 4 | verify | AppOut | ADSB_AIRCRAFT_DATA_APPOUT_PARAMETER S(x) | | Verify that the content of the information output by the station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_High_Resolution_A | | | | | | |
|---|------|--------|-------|---|-----|---|
| Purpose: To demonstrate that a station will broadcast a high resolution sync burst with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:={P,Q,R,S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | AppIn | REQUEST TO TRANSMIT A HIGH RESOLUTION SYNC BURST (ADSB_HIGH_RESOLUTION_APPIN_PARAMETERS(x)) | | Send an instruction at the AppIn PCO to the station under test to transmit a high resolution sync burst. |
| | 3 | await | RF | BURST (s:= add_A) broadcast by station A | | Wait for the message to be broadcast by the station under test. |
| | 4 | verify | RF | BURST (s:= add_A) has the format of "SYNC_HIGH_RESOLUTION_a" | SRa | Verify that the transmitted burst conforms to "SYNC_HIGH_RESOLUTION_a" as set out in the burst format clause. |
| | 5 | verify | RF | ADSB_HIGH_RESOLUTION_RF_OUT_PARAMETERS(x) | | Verify that the parameters have been correctly encoded in the message. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: ADSB_High_Resolution_Rec_A | | | | | | |
|--|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received high resolution sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 sets of parameters. |
| | 2 | send | RF | SYNC_HIGH_RESOLUTION_a (ADSB_HIGH_RESOLUTION_RF_IN_PARAMETERS(x)) | SRa | Send a high resolution sync burst to the station under test from a simulated station B. |
| | 3 | await | AppOut | SYNC_HIGH_RESOLUTION_a MESSAGE INFORMATION | SRa | Wait for the message information to be output by the station under test at the AppOut PCO. |
| | 4 | verify | AppOut | ADSB_HIGH_RESOLUTION_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by the station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Dest_Count | | | | |
|------------------|--------|---|-------------|---|-------------------------|--|
| Purpose: | | To demonstrate that if the destination count (dc) in the header block is zero, then the channel management parameter will apply to all stations and the receiving station will operate in directed rate mode. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 2 | | do_val:= {0, 10} | | Repeat for both zero and non-zero values of do. |
| | 3 | rep x | RF | x:= {2, 1, 0, 3} | | Repeat for different values of txd. |
| | 4 | send | RF | CTRL_CHAN_a (s:= add_G; HEADER: dt:= 5; do:= do_val; dc:= 0; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= x) | Ca | Send a channel management message from a simulated ground station G to the station under test with dc = 0, requesting transmission of a sync burst 6 times per minute for 5 minutes. |
| | 5 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 6 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 7 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 8 | verify | RF | SYNC BURST (s = add_A) transmitted before time:= ctime + 60 | | Verify that the first sync burst transmitted in accordance with the channel management command is transmitted within 4 500 slots. |
| | 9 | verify | RF | IF x = 0 THEN SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 IF x = 1 THEN SYNC BURST = SYNC_BASIC_a and b/g = 1, br/gr = 1 IF x = 2 THEN SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 IF x = 3 THEN SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SBa, SPa, SDa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 10 | rep 29 | | n:=1 | | Start loop. |
| | 11 | verify | RF | SYNC BURST (s = add_A) transmitted before time:= ctime + n x 10 | | Verify that a sync burst is transmitted by the station under test within the requested time interval. |
| | 12 | verify | RF | IF x = 0 THEN SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 IF x = 1 THEN SYNC BURST = SYNC_BASIC_a and b/g = 1, br/gr = 1 IF x = 2 THEN SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 IF x = 3 THEN SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SBa, SPa, SDa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 13 | endrep | | n:= n + 1 | | End loop. |
| | 14 | endrep | | next x | | Go to next value of txd. |
| 15 | endrep | | next do_val | | Go to next value of do. | |
| postamble | 16 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Dir_Rate_A | | | | |
|-----------------|------|---|-----|--|----------|--|
| Purpose: | | To demonstrate that a mobile station receiving a channel management parameter with the directed offset (do) equal to zero in the header block and/or with the individual offset (ido) set equal to zero in a destination block with the station's address, with styp = 0, will operate in directed rate mode. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 3 | | do_val, ido_val:= {0, 0}; {0, 10}; {10, 0} | | Repeat for different do, ido combinations. |
| | 3 | rep 4 | RF | x:= {2, 3} | | Repeat for different values of txd. |
| | 4 | send | RF | CTRL_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= do_val; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: ido:= ido_val; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= x) | Cb | Send a channel management message from a simulated ground station G to the station under test with dc = 0, requesting transmission of a sync burst 6 times per minute for 5 minutes. |
| | 5 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 6 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 7 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 8 | verify | RF | SYNC BURST (s = add_A) transmitted before time:= ctime + 60 | | Verify that the first sync burst transmitted in accordance with the channel management command is transmitted within 4 500 slots. |
| | 9 | verify | RF | IF x = 2 THEN SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 IF x = 3 THEN SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SPa, SDa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 10 | verify | RF | po = 0, pt = 3 in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = 3. |
| | 11 | rep 29 | | n:=1 | | Start loop. |
| | 12 | verify | RF | SYNC BURST (s = add_A) transmitted with nominal slot at time:= sync_time + n x 10 | | Verify that each successive sync burst is transmitted by the station under test with nominal slot at the requested time. |
| | 13 | verify | RF | IF x = 2 THEN SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 IF x = 3 THEN SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SPa, SDa | Verify that the sync bursts are of the expected type and have the b/g and br/gr flags set appropriately. |
| | 14 | verify | RF | po = 0, pt = min (3, TV11 - 1) in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = min (3, TV11 - 1). |
| | 15 | endrep | | n:= n + 1 | | End loop. |
| | 16 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 17 | verify | | SYNC BURST (s = add_A) = SYNC_BURST_c (s = add_A) | Sc | Verify the transmitted burst is now an autonomous sync burst. |
| | 18 | record | RF | sync_time(0):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 19 | rep 60 | | n:= 1 | | Start loop. |
| | 20 | await | | SYNC_BURST_c (s = add_A) | Sc | Wait for a sync burst from the station under test. |
| | 21 | record | RF | sync_time(n):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |

| Test Case Name: Channel_Dir_Rate_A | | | | | | |
|--|------|--------|-----|--|-----|---|
| Purpose: To demonstrate that a mobile station receiving a channel management parameter with the directed offset (do) equal to zero in the header block and/or with the individual offset (ido) set equal to zero in a destination block with the station's address, with styp = 0, will operate in directed rate mode. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 22 | endrep | | n:= n + 1 | | End loop. |
| | 23 | verify | | sync_time(60) - sync_time(0) = 5 minutes | | Verify that the station under test has reverted to autonomous transmission at the default ADS-B rate of 12 sync bursts per minute averaged over the two GSCs. |
| | 24 | endrep | | next x | | Go to next value of txd. |
| | 25 | endrep | | next do_val, ido_val | | Go to next values of do and ido. |
| postamble | 26 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Dir_Rate_B | | | | | | |
|--|------|--------|-----|--|-----|---|
| Purpose: To demonstrate that a mobile station receiving a channel management parameter with the directed offset (do) equal to zero in the header block and/or with the individual offset (ido) set equal to zero in a destination block with the station's address, with styp = 1, will operate in directed rate mode, with transmissions sharing streams in successive minutes. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 3 | | do_val, ido_val:= {0, 0}; {0, 10}; {10, 0} | | Repeat for different do, ido combinations. |
| | 3 | send | RF | CTRL_CHAN_c (s:= add_G; HEADER: dt:= 6; do:= do_val; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 1; sr:= 0; pc:= 0; sc:= 0; DEST: ido:= ido_val; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq(1):= 0; nsd(1):= 0; txd(1):= 2; frq(2):= 0; nsd(2):= 0; txd(2):=3) | Cc | Send a channel management message from a simulated ground station G to the station under test with dc = 0, requesting transmission of a sync burst once per minute for 6 minutes. |
| | 4 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 5 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 6 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 7 | verify | RF | SYNC BURST (s = add_A) transmitted before time:= ctime + 60 | | Verify that the first sync burst transmitted in accordance with the channel management command is transmitted within 4 500 slots. |
| | 8 | verify | RF | SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 | SPa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 9 | verify | RF | po = 0, pt = 3 in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = 3. |
| | 10 | rep 5 | | n:=1 | | Start loop. |
| | 11 | verify | RF | SYNC BURST (s = add_A) transmitted with nominal slot at time:= sync_time + n × 60 | | Verify that each successive sync burst is transmitted by the station under test with nominal slot at the requested time. |

| Test Case Name: Channel_Dir_Rate_B | | | | | | |
|--|------|--------|-----|--|----------|--|
| Purpose: To demonstrate that a mobile station receiving a channel management parameter with the directed offset (do) equal to zero in the header block and/or with the individual offset (ido) set equal to zero in a destination block with the station's address, with styp = 1, will operate in directed rate mode, with transmissions sharing streams in successive minutes. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 12 | verify | RF | IF n = 2 or 4 THEN SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 IF n = 1 or 3 or 5 THEN SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SPa, SDa | Verify that the sync bursts are of the expected type, with the variable part of the sync burst alternating in alternate superframes for the same stream, and that they have the b/g and br/gr flags set appropriately. |
| | 13 | verify | RF | po = 0, pt = min (3, TV11 - 1) in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = min (3, TV11 - 1). |
| | 14 | endrep | | n:= n + 1 | | End loop. |
| | 15 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 16 | verify | | SYNC BURST (s = add_A) = SYNC_BURST_c (s = add_A) | Sc | Verify the transmitted burst is now an autonomous sync burst. |
| | 17 | record | RF | sync_time(0):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 18 | rep 60 | | n:= 1 | | Start loop. |
| | 19 | await | | SYNC_BURST_c (s = add_A) | Sc | Wait for a sync burst from the station under test. |
| | 20 | record | RF | sync_time(n):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 21 | endrep | | n:= n + 1 | | End loop. |
| | 22 | verify | | sync_time(60) - sync_time(0) = 5 minutes | | Verify that the station under test has reverted to autonomous transmission at the default ADS-B rate of 12 sync bursts per minute averaged over the two GSCs. |
| | 23 | endrep | | next do_val, ido_val | | Go to next values of do and ido. |
| postamble | 24 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Address | | | | | | |
|---|------|--------|-----|--|-----|--|
| Purpose: To demonstrate that if none of the addresses matches the receiver, then further processing of the channel management parameter will terminate. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 3 | | n:= 1 | | Start loop. |
| | 3 | await | RF | SYNC_BURST (s:=add_A) | | Wait for an autonomous sync burst from the station under test. |
| | 4 | verify | RF | SYNC_BURST (s:=add_A) is not equal to SYNC_AIRCRAFT_DATA_a | | Verify that the transmitted sync burst does not have the aircraft data variable field. |
| | 5 | endrep | | n:= n + 1 | | End loop. |

| Test Case Name: | | Channel_Address | | | | |
|---|------|-----------------|-----|--|-----|--|
| Purpose: To demonstrate that if none of the addresses matches the receiver, then further processing of the channel management parameter will terminate. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 6 | send | RF | CTRL_CHAN_d (s:= add_G; HEADER: dt:= 5; do:= 0; dc:=2; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: d(1):= add_B; d(2):= add_C; ido(1):= 0; ido(2):= 3; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 3) | Cd | Send a channel management message from simulated ground station G to the station under test having two destination addresses which do not match that of the station under test. |
| | 7 | rep 3 | | n:= 1 | | Start loop. |
| | 8 | await | RF | SYNC_BURST (s:=add_A) | | Wait for a sync burst from the station under test. |
| | 9 | verify | RF | SYNC_BURST (s:=add_A) is not equal to SYNC_AIRCRAFT_DATA_a | | Verify that the transmitted sync burst does not have the aircraft data variable field, and therefore that the station under test has not processed the channel management parameter. |
| | 10 | endrep | | n:= n + 1 | | End loop. |
| postamble | | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Directed | | | | |
|---|------|------------------|-----|--|---------------------|---|
| Purpose: To demonstrate that if the directed offset (do) is equal to a non-zero value in the header block and the individual offset (ido) is set to a non-zero value in a destination block with the station's address, then the addressed stations will operate in directed slot mode. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | RF | x:= {0, 1} | | Repeat for different values of txd. |
| | 3 | send | RF | CTRL_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 100; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: ido:= 15; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= x) | Cb | Send a channel management message from simulated ground station G to the station under test with non-zero values of do and ido. |
| | 4 | record | RF | ctime:= time when the station under test receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 5 | await | | time:= ctime + (100 + 15) x 60/M1 | | Wait for the first sync burst from the station under test. |
| | 6 | verify | RF | SYNC BURST (s = add_A) transmitted in slot beginning at time:= ctime + (100 + 15) x 60/M1 | | Verify that the first sync burst is transmitted by the station under test at the requested time. |
| | 7 | verify | RF | IF x = 0 THEN SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 IF x = 1 THEN SYNC BURST = SYNC_BASIC_a and b/g = 1, br/gr = 1 | SBa, SPa, SDa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 8 | verify | RF | a/d = 1 in SYNC BURST (s = add_A) | | Verify that the autonomous/directed flag is set to 1. |
| | 9 | verify | RF | po = 0, pt = 3 in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = 3. |
| | 10 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |

| Test Case Name: | | Channel_Directed | | | | |
|------------------|------|--|-----|--|---------------|---|
| Purpose: | | To demonstrate that if the directed offset (do) is equal to a non-zero value in the header block and the individual offset (ido) is set to a non-zero value in a destination block with the station's address, then the addressed stations will operate in directed slot mode. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 11 | rep 29 | | n:=1 | | Start loop. |
| | 12 | verify | RF | SYNC BURST (s = add_A) transmitted in slot beginning at time:= sync_time + n x 10 | | Verify that each successive sync burst is transmitted by the station under test at the requested time. |
| | 13 | verify | RF | IF x = 0 THEN SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 IF x = 1 THEN SYNC BURST = SYNC_BASIC_a and b/g = 1, br/gr = 1 | SBa, SPa, SDa | Verify that the sync bursts are of the expected type and have the b/g and br/gr flags set appropriately. |
| | 14 | verify | RF | po = 0, pt = min (3, TV11 - 1) in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = min (3, TV11 - 1). |
| | 15 | verify | RF | a/d = 1 in SYNC BURST (s = add_A) | | Verify that the autonomous/directed flag is set to 1. |
| | 16 | endrep | | n:= n + 1 | | End loop. |
| | 17 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 18 | verify | | SYNC BURST (s = add_A) = SYNC_BURST_c (s = add_A) | Sc | Verify the transmitted burst is now an autonomous sync burst. |
| | 19 | record | RF | sync_time(0):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 20 | rep 60 | | n:= 1 | | Start loop. |
| | 21 | await | | SYNC_BURST_c (s = add_A) | Sc | Wait for a sync burst from the station under test. |
| | 22 | record | RF | sync_time(n):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 23 | endrep | | n:= n + 1 | | End loop. |
| | 24 | verify | | sync_time(60) - sync_time(0) = 5 minutes | | Verify that the station under test has reverted to autonomous transmission at the default ADS-B rate of 12 sync bursts per minute averaged over the two GSCs. |
| | 25 | endrep | | next x | | Go to next value of txd. |
| postamble | 26 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Multicast_A | | | | | | |
|---|------|--------|-----|---|-----|--|
| Purpose: To demonstrate that stations operating in directed-slot mode will not process regional multicast commands. | | | | | | |
| 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_CHAN_b (s:= add_G; HEADER: dt:= 10; do:= 100; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: ido:= 15; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 2) | Cb | Send a directed channel management message from a simulated ground station G to the station under test with an instruction to use directed slots. |
| | 3 | wait | | 2 mins | | Wait for two superframes. |
| | 4 | send | RF | CTRL_CHAN_e (s:= add_G; HEADER: dt:= 10; do:= 0; dc:= 1; csid:= 2; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: vc:= 2; lat:= CPR_LAT(N 12 NM); lat4:= CPR_LAT4(N 12 NM); lon:= CPR_LON(E 12 NM); lon4:= CPR_LON4(E 12 NM); lalt:= 146; ualt:= 162; r ₁ := 10; d ₁ := 1; r ₂ := 15; d ₂ := 59; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 0) | Ce | Send an autotune channel management message from simulated ground station G to the station under test with a regional multicast command. |
| | 5 | await | RF | SYNC BURST (s = add_A) | Sb | Wait for a sync burst from the station under test. |
| | 6 | verify | | SYNC BURST (s = add_A) contains FULL POSITION variable field | | Verify that the station under test is transmitting sync bursts according to the initial channel management command with a directed request and therefore it has not processed the second channel management parameter. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Multicast_B | | | | | | |
|---|------|--------|----------|--|-----|--|
| Purpose: To demonstrate that a mobile station will act on a regional multicast channel management CTRL if it is in the correct region (according to Condition 1), and that processing of the channel management CTRL will terminate if it is not. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 3 | | n:= 1 | | Start loop. |
| | 3 | await | RF | SYNC_BURST (s:=add_A) | | Wait for an autonomous sync burst from the station under test. |
| | 4 | verify | RF | SYNC_BURST (s:=add_A) is not equal to SYNC_AIRCRAFT_DATA_a | | Verify that the transmitted sync burst does not have the aircraft data variable field. |
| | 5 | endrep | | n:= n + 1 | | End loop. |
| | 6 | send | Altitude | Input current altitude as alt:= 34,500 ft | | Input the station's altitude to the station under test, such that the altitude will be outside the altitude limits of the channel management parameter to be sent in step 8. |
| | 7 | send | Position | Input current position as lat:= N 12.5 NM; lon:= E 41 NM | | Input the station's position to the station under test, such that the position will be within the position limits of the channel management parameter to be sent in step 8. |

| Test Case Name: | | Channel_Multicast_B | | | | |
|-----------------|------|--|----------|--|-----|--|
| Purpose: | | To demonstrate that a mobile station will act on a regional multicast channel management CTRL if it is in the correct region (according to Condition 1), and that processing of the channel management CTRL will terminate if it is not. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 8 | send | RF | CTRL_CHAN_e (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: vc:= 2; lat:= CPR_LAT(N 12 NM); lat4:= CPR_LAT4(N 12 NM); lon:= CPR_LON(E 12 NM); lon4:= CPR_LON4(E 12 NM); lalt:= 146; ualt:= 162; r ₁ := 10; d ₁ := 59; r ₂ := 64; d ₂ := 1; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 3) | Ce | Send an autotune channel management message from simulated ground station G to the station under test with a regional multicast command. The command requests 6 sync burst per minute of type aircraft data. |
| | 9 | rep 3 | | n:= 1 | | Start loop. |
| | 10 | await | RF | SYNC_BURST (s:=add_A) | | Wait for a sync burst from the station under test. |
| | 11 | verify | RF | SYNC_BURST (s:=add_A) is not equal to SYNC_AIRCRAFT_DATA_a | | Verify that the transmitted sync burst does not have the aircraft data variable field, and therefore that the station under test has not acted on the channel management parameter. |
| | 12 | endrep | | n:= n + 1 | | End loop. |
| | 13 | send | Altitude | Input current altitude as alt:= 35,500 ft | | Input the station's altitude to the station under test, such that the altitude will be within the altitude limits of the channel management parameter to be sent in step 15. |
| | 14 | send | Position | Input current position as lat:= N 12.5 NM; lon:= E 41 NM | | Input the station's position to the station under test, such that the position will be within the position limits of the channel management parameter to be sent in step 15. |
| | 15 | send | RF | CTRL_CHAN_e (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 1; csid:= 2; tc:= 0; fc:= 1; styp:= 0; sr:= 5; pc:= 0; sc:= 0; DEST: vc:= 2; lat:= CPR_LAT(N 12 NM); lat4:= CPR_LAT4(N 12 NM); lon:= CPR_LON(E 12 NM); lon4:= CPR_LON4(E 12 NM); lalt:= 146; ualt:= 162; r ₁ := 10; d ₁ := 59; r ₂ := 64; d ₂ := 1; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 3) | Ce | Send an autotune channel management message from simulated ground station G to the station under test with a regional multicast command. The command requests 5 sync burst per minute of type aircraft data. |
| | 16 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 17 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 18 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 19 | verify | RF | SYNC BURST (s = add_A) transmitted before time:= ctime + 60 | | Verify that the first sync burst transmitted in accordance with the channel management command is transmitted within 4 500 slots. |
| | 20 | verify | RF | SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SDa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 21 | rep 5 | | n:=1 | | Start loop. |

| Test Case Name: Channel_Multicast_B | | | | | | |
|---|------|--------|-----|--|-----|--|
| Purpose: To demonstrate that a mobile station will act on a regional multicast channel management CTRL if it is in the correct region (according to Condition 1), and that processing of the channel management CTRL will terminate if it is not. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 22 | verify | RF | SYNC BURST (s = add_A) transmitted with nominal slot at time:= sync_time + n × 12 | | Verify that each successive sync burst is transmitted by the station under test with nominal slot at the requested time. |
| | 23 | verify | RF | SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SDa | Verify that the sync bursts are of the expected type and have the b/g and br/gr flags set appropriately. |
| | 24 | endrep | | n:= n + 1 | | End loop. |
| postamble | 25 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Multicast_C | | | | | | |
|--|------|--------|----------|--|-----|--|
| Purpose: To demonstrate that a mobile station will act on a regional multicast channel management CTRL if it is in the correct region (according to Condition 2a), and that processing of the channel management CTRL will terminate if it is not. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 3 | | n:= 1 | | Start loop. |
| | 3 | await | RF | SYNC_BURST (s:=add_A) | | Wait for an autonomous sync burst from the station under test. |
| | 4 | verify | RF | SYNC_BURST (s:=add_A) is not equal to SYNC_AIRCRAFT_DATA_a | | Verify that the transmitted sync burst does not have the aircraft data variable field. |
| | 5 | endrep | | n:= n + 1 | | End loop. |
| | 6 | send | Altitude | Input current altitude as alt:= 35,500 ft | | Input the station's altitude to the station under test, such that the altitude will be within the altitude limits of the channel management parameter to be sent in step 8. |
| | 7 | send | Position | Input current position as lat:= N 12.5 NM; lon:= E 42 NM | | Input the station's position to the station under test, such that the position will be outside the position limits of the channel management parameter to be sent in step 8. |
| | 8 | send | RF | CTRL_CHAN_e (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: vc:= 2; lat:= CPR_LAT(N 12 NM); lat4:= CPR_LAT4(N 12 NM); lon:= CPR_LON(E 12 NM); lon4:= CPR_LON4(E 12 NM); lalt:= 146; ualt:= 162; r ₁ := 10; d ₁ := 59; r ₂ := 64; d ₂ := 1; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 3) | Ce | Send an autotune channel management message from simulated ground station G to the station under test with a regional multicast command. The command requests 6 sync burst per minute of type aircraft data. |
| | 9 | rep 3 | | n:= 1 | | Start loop. |
| | 10 | await | RF | SYNC_BURST (s:=add_A) | | Wait for a sync burst from the station under test. |
| | 11 | verify | RF | SYNC_BURST (s:=add_A) is not equal to SYNC_AIRCRAFT_DATA_a | | Verify that the transmitted sync burst does not have the aircraft data variable field, and therefore that the station under test has not acted on the channel management parameter. |

| Test Case Name: | | Channel_Multicast_C | | | | |
|------------------|------|---|----------|--|-----|--|
| Purpose: | | To demonstrate that a mobile station will act on a regional multicast channel management CTRL if it is in the correct region (according to Condition 2a), and that processing of the channel management CTRL will terminate if it is not. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 12 | endrep | | n:= n + 1 | | End loop. |
| | 13 | send | Altitude | Input current altitude as alt:= 35,500 ft | | Input the station's altitude to the station under test, such that the altitude will be within the altitude limits of the channel management parameter to be sent in step 15. |
| | 14 | send | Position | Input current position as lat:= N 12.5 NM; lon:= E 41 NM | | Input the station's position to the station under test, such that the position will be within the position limits of the channel management parameter to be sent in step 15. |
| | 15 | send | RF | CTRL_CHAN_e (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 1; csid:= 2; tc:= 0; fc:= 1; styp:= 0; sr:= 5; pc:= 0; sc:= 0; DEST: vc:= 2; lat:= CPR_LAT(N 12 NM); lat4:= CPR_LAT4(N 12 NM); lon:= CPR_LON(E 12 NM); lon4:= CPR_LON4(E 12 NM); lalt:= 146; ualt:= 162; r ₁ := 10; d ₁ := 59; r ₂ := 64; d ₂ := 1; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 3) | Ce | Send an autotune channel management message from simulated ground station G to the station under test with a regional multicast command. The command requests 5 sync burst per minute of type aircraft data. |
| | 16 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 17 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 18 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 19 | verify | RF | SYNC BURST (s = add_A) transmitted before time:= ctime + 60 | | Verify that the first sync burst transmitted in accordance with the channel management command is transmitted within 4 500 slots. |
| | 20 | verify | RF | SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SDa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 21 | rep 5 | | n:=1 | | Start loop. |
| | 22 | verify | RF | SYNC BURST (s = add_A) transmitted with nominal slot at time:= sync_time + n x 12 | | Verify that each successive sync burst is transmitted by the station under test with nominal slot at the requested time. |
| | 23 | verify | RF | SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SDa | Verify that the sync bursts are of the expected type and have the b/g and br/gr flags set appropriately. |
| | 24 | endrep | | n:= n + 1 | | End loop. |
| postamble | 25 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Multicast_D | | | | |
|-----------------|------|---|----------|--|-----|--|
| Purpose: | | To demonstrate that a mobile station will act on a regional multicast channel management CTRL if it is in the correct region (according to Condition 2b), and that processing of the channel management CTRL will terminate if it is not. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 3 | | n:= 1 | | Start loop. |
| | 3 | await | RF | SYNC_BURST (s:=add_A) | | Wait for an autonomous sync burst from the station under test. |
| | 4 | verify | RF | SYNC_BURST (s:=add_A) is not equal to SYNC_AIRCRAFT_DATA_a | | Verify that the transmitted sync burst does not have the aircraft data variable field. |
| | 5 | endrep | | n:= n + 1 | | End loop. |
| | 6 | send | Altitude | Input current altitude as alt:= 35,500 ft | | Input the station's altitude to the station under test, such that the altitude will be within the altitude limits of the channel management parameter to be sent in step 8. |
| | 7 | send | Position | Input current position as lat:= N 42 NM; lon:= E 12.5 NM | | Input the station's position to the station under test, such that the position will be outside the position limits of the channel management parameter to be sent in step 8. |
| | 8 | send | RF | CTRL_CHAN_e (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: vc:= 2; lat:= CPR_LAT(N 12 NM); lat4:= CPR_LAT4(N 12 NM); lon:= CPR_LON(E 12 NM); lon4:= CPR_LON4(E 12 NM); lalt:= 146; ualt:= 162; r ₁ := 10; d ₁ := 1; r ₂ := 64; d ₂ := 59; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 3) | Ce | Send an autotune channel management message from simulated ground station G to the station under test with a regional multicast command. The command requests 6 sync burst per minute of type aircraft data. |
| | 9 | rep 3 | | n:= 1 | | Start loop. |
| | 10 | await | RF | SYNC_BURST (s:=add_A) | | Wait for a sync burst from the station under test. |
| | 11 | verify | RF | SYNC_BURST (s:=add_A) is not equal to SYNC_AIRCRAFT_DATA_a | | Verify that the transmitted sync burst does not have the aircraft data variable field, and therefore that the station under test has not acted on the channel management parameter. |
| | 12 | endrep | | n:= n + 1 | | End loop. |
| | 13 | send | Altitude | Input current altitude as alt:= 35,500 ft | | Input the station's altitude to the station under test, such that the altitude will be within the altitude limits of the channel management parameter to be sent in step 15. |
| | 14 | send | Position | Input current position as lat:= N 41 NM; lon:= E 12.5 NM | | Input the station's position to the station under test, such that the position will be within the position limits of the channel management parameter to be sent in step 15. |

| Test Case Name: | | Channel_Multicast_D | | | | |
|------------------|------|---|-----|--|-----|--|
| Purpose: | | To demonstrate that a mobile station will act on a regional multicast channel management CTRL if it is in the correct region (according to Condition 2b), and that processing of the channel management CTRL will terminate if it is not. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 15 | send | RF | CTRL_CHAN_e (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 1; csid:= 2; tc:= 0; fc:= 1; styp:= 0; sr:= 5; pc:= 0; sc:= 0; DEST: vc:= 2; lat:= CPR_LAT(N 12 NM); lat4:= CPR_LAT4(N 12 NM); lon:= CPR_LON(E 12 NM); lon4:= CPR_LON4(E 12 NM); lalt:= 146; ualt:= 162; r ₁ := 10; d ₁ := 1; r ₂ := 64; d ₂ := 59; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 3) | Ce | Send an autotune channel management message from simulated ground station G to the station under test with a regional multicast command. The command requests 5 sync burst per minute of type aircraft data. |
| | 16 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 17 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 18 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 19 | verify | RF | SYNC BURST (s = add_A) transmitted before time:= ctime + 60 | | Verify that the first sync burst transmitted in accordance with the channel management command is transmitted within 4 500 slots. |
| | 20 | verify | RF | SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SDa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 21 | rep 5 | | n:=1 | | Start loop. |
| | 22 | verify | RF | SYNC BURST (s = add_A) transmitted with nominal slot at time:= sync_time + n × 12 | | Verify that each successive sync burst is transmitted by the station under test with nominal slot at the requested time. |
| | 23 | verify | RF | SYNC BURST = SYNC_AIRCRAFT_DATA_a and b/g = 0 | SDa | Verify that the sync bursts are of the expected type and have the b/g and br/gr flags set appropriately. |
| | 24 | endrep | | n:= n + 1 | | End loop. |
| postamble | 25 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Invalid | | | | |
|------------------|------|---|-----|--|-----|---|
| Purpose: | | To demonstrate that if styp = 1 and NES is not a multiple of (2 + sr), then a channel management CTRL will be judged invalid, and the entire channel management parameter will be discarded without further processing. | | | | |
| 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_CHAN_b (s:= add_G; HEADER: dt:= 11; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 1; sr:= 0; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 3) | Cb | Send a channel management message from simulated ground station G to the station under test with styp = 1 and NES not a multiple of (2 + sr). |
| | 3 | rep 60 | | n:= 1 | | Start loop. |
| | 4 | await | RF | SYNC BURST (s = add_A) | Sb | Wait for a sync burst from the station under test. |
| | 5 | record | RF | time(n):= time at the beginning of slot containing SYNC BURST (s = add_A) | | Record the time that the message was received. |
| | 6 | endrep | | n:= n + 1 | | End loop. |
| | 7 | verify | | time(60) - time(1) = 5 mins | | Verify that the station under test does not transmit according to the instructions of the channel management parameter, but continues autonomous behaviour on the GSCs (i.e. 12 messages in 1 min). |
| postamble | 8 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Frequency | | | | | | |
|--|------|--------|-----|---|-----|---|
| Purpose: To demonstrate that if all of the elements that share a stream are not transmitted on the same frequency in a channel management CTRL, then the message will be judged invalid and the entire CTRL will be discarded. | | | | | | |
| 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_CHAN_f (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 1; sr:= 0; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep(1):= 0; auto(1):= 0; rcv(1):= 0; f(1):= transmit freq; sleep(2):= 0; auto(2):= 0; rcv(2):= 0; f(2):= 135.000 MHz; SCRIPT: frq(1):= 0; nsd(1):= 0; txd(1):= 2; frq(2):= 0; nsd(2):= 0; txd(2):= 3) | Cf | Send a channel management message from simulated ground station G to the station under test, with different frequencies for different elements of a stream (set transmit frequency not equal to 135.000 MHz). |
| | 3 | rep 60 | | n:= 1 | | Start loop. |
| | 4 | await | RF | SYNC BURST (s = add_A) | Sb | Wait for a sync burst from the station under test. |
| | 5 | record | RF | time(n):= time at the beginning of slot containing SYNC BURST (s = add_A) | | Record the time that the message was received. |
| | 6 | endrep | | n:= n + 1 | | End loop. |
| | 7 | verify | | time(60) - time(1) = 5 mins | | Verify that the station under test does not transmit according to the instructions of the channel management parameter, but continues autonomous behaviour on the GSCs (i.e. 12 messages in 1 min). |
| postamble | 8 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Length | | | | | | |
|---|------|--------|-----|---|-----|---|
| Purpose: To demonstrate that if all of the elements that share a stream are not of the same length in a channel management CTRL, then a basic sync burst will be transmitted for all elements in that stream. | | | | | | |
| 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_CHAN_g (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 0; csid:= 1; tc:= 1; fc:= 1; styp:= 1; sr:= 0; pc:= 0; sc:= 0; TRANS: id:= 8hex; aux:= 0; b/g:= 0; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq(1):= 0; nsd(1):= 0; txd(1):= 2; frq(2):= 0; nsd(2):= 1; txd(2):=0) | Cg | Send a channel management message from simulated ground station G to the station under test with not all the elements of the stream having the same length. |
| | 3 | rep 6 | | n:= 1 | | Start loop. |
| | 4 | await | RF | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 6 | verify | | SYNC BURST (s = add_A) contains BASIC variable part | | Verify that a basic sync burst is emitted by the station under test for all elements in the stream. |
| | 7 | endrep | | n:= n + 1 | | End loop. |
| postamble | 8 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Plea | | | | |
|-----------------|------|--|-----|---|-----|---|
| Purpose: | | To demonstrate that if a station receives a plea request while transmitting according to a channel management CTRL, it shall use NTM as the default reporting rate for the frequency to construct the plea response. | | | | |
| 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_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 3; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= x) | Cb | Send a channel management message from simulated ground station G to the station under test with styp = 0 and NTM = 3, giving three streams per superframe. |
| | 3 | rep 3 | | n:= 1 | | Start loop. |
| | 4 | await | RF | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 5 | record | | sync_time(n) := time at the beginning of slot containing SYNC BURST (s =add_A) | | Record the time that the sync burst was transmitted. |
| | 6 | endrep | | n:= n + 1 | | End loop. |
| | 7 | send | RF | PLEA_a (s:= add_B, d:= add_A) | Pa | Send a plea request to the station under test from a simulated station B. |
| | 8 | await | RF | PLEA_RESP_a (s:=add_A) | PRa | Wait for a plea response burst from the station under test. |
| | 9 | record | RF | plea_time:= time at beginning of slot containing PLEA_RESP_a (s =add_A) | PRa | Record the time that the plea response was transmitted. |
| | 10 | record | RF | OFF:= off NR:= nr A1:= a1 A2:= a2 A11:= a11 | | Record the values contained in the plea response. |
| | 11 | verify | RF | NR = 3 | | Verify that the nominal update rate equates to three per minute. |
| | 12 | verify | RF | A4 to A11 set to zero | | Verify that the unused additional slots are set to zero. |
| | 13 | verify | RF | sync_time(x1) + 60 = plea_time + off × 60/M1 + 20 + A1 sync_time(x2) + 60 = plea_time + off × 60/M1 + 40 + A2 sync_time(x3) + 60 = plea_time + off × 60/M1 + 60 + A3 where (x1, x2, x3) = (1, 2, 3) or (2, 3, 1) or (3, 1, 2) | | Verify that the slots reserved in the plea response correspond to the slots in the streams set up by the channel management message. |
| | 14 | wait | | 1 minute | | |
| | 15 | send | RF | CTRL_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 1; csid:= 2; tc:= 0; fc:= 1; styp:= 0; sr:= 4; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= x) | Cb | Send a channel management message from simulated ground station G to the station under test with styp = 0 and NTM = 4, giving four streams per superframe. |
| | 16 | rep 4 | | n:= 1 | | Start loop. |
| | 17 | await | RF | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 18 | record | | sync_time(n) := time at the beginning of slot containing SYNC BURST (s =add_A) | | Record the time that the sync burst was transmitted. |

| Test Case Name: | | Channel_Plea | | | | |
|------------------|------|--|-----|---|-----|--|
| Purpose: | | To demonstrate that if a station receives a plea request while transmitting according to a channel management CTRL, it shall use NTM as the default reporting rate for the frequency to construct the plea response. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 19 | endrep | | n:= n + 1 | | End loop. |
| | 20 | send | RF | PLEA_a (s:= add_B, d:= add_A) | Pa | Send a plea request to the station under test from a simulated station B. |
| | 21 | await | RF | PLEA_RESP_a (s:=add_A) | PRa | Wait for a plea response burst from the station under test. |
| | 22 | record | RF | plea_time:= time at beginning of slot containing PLEA_RESP_a (s =add_A) | PRa | Record the time that the plea response was transmitted. |
| | 23 | record | RF | OFF:= off NR:= nr A1:= a1 A2:= a2 A11:= a11 | | Record the values contained in the plea response. |
| | 24 | verify | RF | NR = 4 | | Verify that the nominal update rate equates to three per minute. |
| | 25 | verify | RF | A5 to A11 set to zero | | Verify that the unused additional slots are set to zero. |
| | 26 | verify | RF | sync_time(x1) + 60 = plea_time + off × 60/M1 + 15 + A1 sync_time(x2) + 60 = plea_time + off × 60/M1 + 30 + A2 sync_time(x3) + 60 = plea_time + off × 60/M1 + 45 + A3 sync_time(x4) + 60 = plea_time + off × 60/M1 + 60 + A4 where (x1, x2, x3, x4) = (1, 2, 3, 4) or (2, 3, 4, 1) or (3, 4, 1, 2) or (4, 1, 2, 3) | | Verify that the slots reserved in the plea response correspond to the slots in the streams set up by the channel management message. |
| postamble | 27 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Cancel_DT | | | | |
|------------------|-----------|---|-----|--|-----|--|
| Purpose: | | To demonstrate that a station, previously in receipt of a channel management parameter, that receives a channel management parameter with the directed timeout subfield set to 15, the directed offset (do) set to the offset from the first slot of the cancellation channel management parameter to the first slot for which the reservation shall be cancelled, and all other subfields set to the same values as in the original channel management CTRL parameter to be cancelled, will revert to default sync burst operations. | | | | |
| 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_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:=2) | Cb | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 5 minutes. |
| | 3 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 4 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 5 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 6 | verify | RF | SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 | SPa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 7 | await | | time:= sync_time + 30 | | |
| | 8 | send | RF | CTRL_CHAN_b (s:= add_G; HEADER: dt:= 15; do:= 2 250; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:=2) in slot beginning at time:= sync_time + 30 | Cb | Send a channel management message from a simulated ground station G to the station under test that is the same as the original channel management parameter but with dt = 15 and with do pointing to the next slot in which the station under test was due to transmit according to the original channel management parameter. |
| | 9 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 10 | verify | | SYNC BURST (s = add_A) = SYNC_BURST_c (s = add_A) | Sc | Verify the transmitted burst is now an autonomous sync burst. |
| | 11 | record | RF | sync_time(0):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 12 | rep 60 | | n:= 1 | | Start loop. |
| | 13 | await | | SYNC_BURST_c (s = add_A) | Sc | Wait for a sync burst from the station under test. |
| | 14 | record | RF | sync_time(n):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 15 | endrep | | n:= n + 1 | | End loop. |
| | 16 | verify | | sync_time(60) - sync_time(0) = 5 minutes | | Verify that the station under test has reverted to autonomous transmission at the default ADS-B rate of 12 sync bursts per minute averaged over the two GSCs. |
| | postamble | 17 | | | | |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Cancel_CSID_A | | | | |
|------------------|-----------|--|-----|---|-----|--|
| Purpose: | | To demonstrate that a station previously in receipt of a channel management parameter that is commanded to cancel all reservations for sync burst transmission by the same channel management parameter but with csid = 15 will revert to default sync burst operations. | | | | |
| 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_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:=2) | Cb | Send a channel management message from a simulated ground station G to the station under test with dc = 0, requesting transmission of a sync burst 6 times per minute for 5 minutes. |
| | 3 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 4 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 5 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 6 | verify | RF | SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 | SPa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 7 | send | RF | CTRL_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 1; csid:= 15; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:=2) | Cb | Send a channel management message from a simulated ground station G to the station under test that is the same as the original channel management parameter but with csid = 15. |
| | 8 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 9 | verify | | SYNC BURST (s = add_A) = SYNC_BURST_c (s = add_A) | Sc | Verify the transmitted burst is now an autonomous sync burst. |
| | 10 | record | RF | sync_time(0):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 11 | rep 60 | | n:= 1 | | Start loop. |
| | 12 | await | | SYNC_BURST_c (s = add_A) | Sc | Wait for a sync burst from the station under test. |
| | 13 | record | RF | sync_time(n):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 14 | endrep | | n:= n + 1 | | End loop. |
| | 15 | verify | | sync_time(60) - sync_time(0) = 5 minutes | | Verify that the station under test has reverted to autonomous transmission at the default ADS-B rate of 12 sync bursts per minute averaged over the two GSCs. |
| | postamble | 16 | | | | |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Cancel_CSID_B | | | | |
|------------------|-----------|---|-----|---|-----|--|
| Purpose: | | To demonstrate that a station previously in receipt of a channel management parameter that is commanded to cancel all reservations for sync burst transmission by a different channel management parameter with csid = 15 will revert to default sync burst operations. | | | | |
| 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_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:=2) | Cb | Send a channel management message from a simulated ground station G to the station under test with dc = 0, requesting transmission of a sync burst 6 times per minute for 5 minutes. |
| | 3 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 4 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 5 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 6 | verify | RF | SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 | SPa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 7 | send | RF | CTRL_CHAN_b (s:= add_G; HEADER: dt:= 7; do:= 0; dc:= 1; csid:= 15; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:=3) | Cb | Send a channel management message from a simulated ground station G to the station under test that has csid = 15 and two other fields that are different from the original channel management parameter. |
| | 8 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 9 | verify | | SYNC BURST (s = add_A) = SYNC_BURST_c (s = add_A) | Sc | Verify the transmitted burst is now an autonomous sync burst. |
| | 10 | record | RF | sync_time(0):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 11 | rep 60 | | n:= 1 | | Start loop. |
| | 12 | await | | SYNC_BURST_c (s = add_A) | Sc | Wait for a sync burst from the station under test. |
| | 13 | record | RF | sync_time(n):= time at beginning of slot containing SYNC_BURST_c (s = add_A) | Sc | Record the time the sync burst was transmitted. |
| | 14 | endrep | | n:= n + 1 | | End loop. |
| | 15 | verify | | sync_time(60) - sync_time(0) = 5 minutes | | Verify that the station under test has reverted to autonomous transmission at the default ADS-B rate of 12 sync bursts per minute averaged over the two GSCs. |
| | postamble | 16 | | | | |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Cancel_Auto | | | | |
|-----------------|------|--|-----|---|-----|---|
| Purpose: | | To demonstrate that if a receiving station is transmitting autonomously the data for which a channel management command was received, then it will cancel its existing reservations and operate in accordance with the parameters of the CTRL command. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 5 | | n:= 1 | | Start loop. |
| | 3 | await | RF | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 4 | verify | RF | SYNC BURST (s = add_A) conforms to SYNC_BURST_c (s:= add_A) | Sc | Verify the burst is an autonomous sync burst. |
| | 5 | record | RF | sync_time(n):= time at beginning of slot containing SYNC_BURST_c (s =add_A) | Sc | Record the positions of autonomous sync bursts transmitted by the station under test. |
| | 6 | verify | | SYNC_BURST_c (s =add_A) contains BASIC variable part | | Verify that the station under test is transmitting the basic variable field. |
| | 7 | endrep | | n:= n + 1 | | End loop. |
| | 8 | await | | time:= sync_time(5) + 5 | | Wait until 5 seconds after the last autonomous sync burst. |
| | 9 | send | RF | CTRL_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 100; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: ido:= 15; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 0) transmitted in slot beginning at time:= sync_time(5) + 5 | Cb | Send a channel management message for directed slots from simulated ground station G to the station under test requesting transmission of sync bursts with the basic variable field in specified slots. |
| | 10 | await | | time:= sync_time(5) + 5 + (100 + 15) x 60/M1 | | Wait for the first sync burst from the station under test to be transmitted as part of the channel management command. |
| | 11 | verify | RF | SYNC BURST (s = add_A) transmitted in slot beginning at time:= sync_time(5) + 5 + (100 + 15) x 60/M1 | | Verify that the first sync burst is transmitted by the station under test at the requested time. |
| | 12 | verify | RF | SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 | SBa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 13 | verify | RF | a/d = 1 in SYNC BURST (s = add_A) | | Verify that the autonomous/directed flag is set to 1. |
| | 14 | verify | RF | po = 0, pt = 3 in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = 3. |
| | 15 | record | RF | dir_sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the directed sync burst was transmitted. |
| | 16 | await | | time:= sync_time(5) + 60 | | Wait for the next slot in which an autonomous sync burst would have been expected in the absence of the channel management command. |
| | 17 | verify | RF | SYNC_BURST_c (s =add_A) transmitted in slot beginning at time:= sync_time(5) + 60 | Sc | Verify that the slot contains an autonomous sync burst. |
| | 18 | verify | | pt = 0 and po = 0 in SYNC_BURST_c (s =add_A) | Sc | Verify that the station under test transmits cancellation sync bursts with pt = 0 and po = 0. |
| | 19 | rep 29 | | n:=1 | | Start loop to check the rest of the sync bursts commanded by the channel management command. |

| Test Case Name: | | Channel_Cancel_Auto | | | | |
|------------------|------|--|-----|---|-----|--|
| Purpose: | | To demonstrate that if a receiving station is transmitting autonomously the data for which a channel management command was received, then it will cancel its existing reservations and operate in accordance with the parameters of the CTRL command. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 20 | verify | RF | SYNC BURST (s = add_A) transmitted in slot beginning at time:= dir_sync_time + n x 10 | | Verify that each successive sync burst is transmitted by the station under test at the requested time. |
| | 21 | verify | RF | SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 | SBa | Verify that the sync bursts are of the expected type and have the b/g and br/gr flags set appropriately. |
| | 22 | verify | RF | po = 0, pt = min (3, TV11 - 1) in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = min (3, TV11 - 1). |
| | 23 | verify | RF | a/d = 1 in SYNC BURST (s = add_A) | | Verify that the autonomous/directed flag is set to 1. |
| | 24 | endrep | | n:= n + 1 | | End loop. |
| postamble | 25 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Cancel_CSID_C | | | | |
|-----------------|------|--|-----|---|-----|---|
| Purpose: | | To demonstrate that a station that receives a channel management CTRL parameter with csid = 0 will terminate any previous default sync burst operations and initiate operations in accordance with the current CTRL parameter. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 5 | | n:= 1 | | Start loop. |
| | 3 | await | RF | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 4 | verify | RF | SYNC BURST (s = add_A) conforms to SYNC_BURST_c (s:= add_A) | Sc | Verify the burst is an autonomous sync burst. |
| | 5 | record | RF | sync_time(n):= time at beginning of slot containing SYNC_BURST_c (s =add_A) | Sc | Record the positions of autonomous sync bursts transmitted by the station under test. |
| | 6 | verify | | SYNC_BURST_c (s =add_A) contains BASIC variable part | | Verify that the station under test is transmitting the basic variable field. |
| | 7 | endrep | | n:= n + 1 | | End loop. |
| | 8 | await | | time:= sync_time(5) + 5 | | Wait until 5 seconds after the last autonomous sync burst. |
| | 9 | send | RF | CTRL_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 100; dc:= 1; csid:= 0; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: ido:= 15; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 2) transmitted in slot beginning at time:= sync_time(5) + 5 | Cb | Send a channel management message for directed slots from simulated ground station G to the station under test with csid = 0. |
| | 10 | await | | time:= sync_time(5) + 5 + (100 + 15) x 60/M1 | | Wait for the first sync burst from the station under test to be transmitted as part of the channel management command. |
| | 11 | verify | RF | SYNC BURST (s = add_A) transmitted in slot beginning at time:= sync_time(5) + 5 + (100 + 15) x 60/M1 | | Verify that the first sync burst is transmitted by the station under test at the requested time. |

| Test Case Name: | | Channel_Cancel_CSID_C | | | | |
|------------------|------|--|-----|---|-----|---|
| Purpose: | | To demonstrate that a station that receives a channel management CTRL parameter with csid = 0 will terminate any previous default sync burst operations and initiate operations in accordance with the current CTRL parameter. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 12 | verify | RF | SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 | SPa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 13 | verify | RF | a/d = 1 in SYNC BURST (s = add_A) | | Verify that the autonomous/directed flag is set to 1. |
| | 14 | verify | RF | po = 0, pt = 3 in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = 3. |
| | 15 | record | RF | dir_sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the directed sync burst was transmitted. |
| | 16 | await | | time:= sync_time(5) + 60 | | Wait for the next slot in which an autonomous sync burst would have been expected in the absence of the channel management command. |
| | 17 | verify | RF | SYNC_BURST_c (s =add_A) transmitted in slot beginning at time:= sync_time(5) + 60 | Sc | Verify that the slot contains an autonomous sync burst. |
| | 18 | verify | | pt = 0 and po = 0 in SYNC_BURST_c (s =add_A) | Sc | Verify that the station under test transmits cancellation sync bursts with pt = 0 and po = 0. |
| | 19 | rep 29 | | n:=1 | | Start loop to check the rest of the sync bursts commanded by the channel management command. |
| | 20 | verify | RF | SYNC BURST (s = add_A) transmitted in slot beginning at time:= dir_sync_time + n x 10 | | Verify that each successive sync burst is transmitted by the station under test at the requested time. |
| | 21 | verify | RF | SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 | SPa | Verify that the sync bursts are of the expected type and have the b/g and br/gr flags set appropriately. |
| | 22 | verify | RF | po = 0, pt = min (3, TV11 - 1) in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = min (3, TV11 - 1). |
| | 23 | verify | RF | a/d = 1 in SYNC BURST (s = add_A) | | Verify that the autonomous/directed flag is set to 1. |
| | 24 | endrep | | n:= n + 1 | | End loop. |
| postamble | 25 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Cancel_CSID_D | | | | |
|-----------------|------|---|-----|---|-----|--|
| Purpose: | | To demonstrate that a station that receives a channel management CTRL parameter with csid matching a previously-received channel management CTRL parameter will terminate any previous operations commanded by the previous CTRL parameter and initiate operations in accordance with the current CTRL parameter. | | | | |
| 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_CHAN_a (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 0; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 3) | Ca | Send a channel management message from a simulated ground station G to the station under test with dc = 0, requesting transmission of a sync burst once per minute for 5 minutes. |
| | 3 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 4 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 5 | verify | RF | SYNC BURST (s = add_A) = SYNC_AIRCRAFT_DATA_a (s = add_A) and b/g = 0 | SDa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 6 | await | | time:= sync_time + 60 | | Wait for the next sync burst. |
| | 7 | verify | RF | SYNC_AIRCRAFT_DATA_a (s = add_A) contained in slot beginning at time:= sync_time + 60 | SDa | Verify that the next sync burst is transmitted in the expected slot. |
| | 8 | await | | time:= sync_time + 65 | | Wait until 5 seconds after the last sync burst. |
| | 9 | send | RF | CTRL_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 100; dc:= 1; csid:= 0; tc:= 0; fc:= 1; styp:= 0; sr:= 6; pc:= 0; sc:= 0; DEST: ido:= 15; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 2) transmitted in slot beginning at time:= sync_time + 65 | Cb | Send a channel management message for directed slots from simulated ground station G to the station under test with csid = 0. |
| | 10 | await | | time:= sync_time + 65 + (100 + 15) × 60/M1 | | Wait for the first sync burst from the station under test to be transmitted as part of the channel management command. |
| | 11 | verify | RF | SYNC BURST (s = add_A) transmitted in slot beginning at time:= sync_time + 65 + (100 + 15) × 60/M1 | | Verify that the first sync burst is transmitted by the station under test at the requested time. |
| | 12 | verify | RF | SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 | SPa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 13 | verify | RF | a/d = 1 in SYNC BURST (s = add_A) | | Verify that the autonomous/directed flag is set to 1. |
| | 14 | verify | RF | po = 0, pt = 3 in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = 3. |
| | 15 | record | RF | dir_sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the directed sync burst was transmitted. |
| | 16 | await | | time:= sync_time + 120 | | Wait for the next slot in which the sync burst commanded by the first channel management command would have been expected in the absence of the second channel management command. |

| Test Case Name: Channel_Cancel_CSID_D | | | | | | |
|--|------|--------|-----|--|-----|--|
| Purpose: To demonstrate that a station that receives a channel management CTRL parameter with csid matching a previously-received channel management CTRL parameter will terminate any previous operations commanded by the previous CTRL parameter and initiate operations in accordance with the current CTRL parameter. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 17 | verify | RF | SYNC_AIRCRAFT_DATA_a (s = add_A) transmitted in slot beginning at time:= sync_time + 120 | SDa | Verify that the slot contains an autonomous sync burst. |
| | 18 | verify | | pt = 0 and po = 0 in SYNC_AIRCRAFT_DATA_a (s = add_A) | SDa | Verify that the station under test transmits cancellation sync bursts with pt = 0 and po = 0. |
| | 19 | rep 29 | | n:=1 | | Start loop to check the rest of the sync bursts commanded by the second channel management command. |
| | 20 | verify | RF | SYNC BURST (s = add_A) transmitted in slot beginning at time:= dir_sync_time + n x 10 | | Verify that each successive sync burst is transmitted by the station under test at the requested time. |
| | 21 | verify | RF | SYNC BURST = SYNC_FULL_POSITION_a and b/g = 0, br/gr = 0 | SPa | Verify that the sync bursts are of the expected type and have the b/g and br/gr flags set appropriately. |
| | 22 | verify | RF | po = 0, pt = min (3, TV11 - 1) in SYNC BURST (s = add_A) | | Verify that the sync burst contains po = 0 and pt = min (3, TV11 - 1). |
| | 23 | verify | RF | a/d = 1 in SYNC BURST (s = add_A) | | Verify that the autonomous/directed flag is set to 1. |
| | 24 | endrep | | n:= n + 1 | | End loop. |
| postamble | 25 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Unrecognize | | | | | | |
|---|------|--------|-----|--|-----|--|
| Purpose: To demonstrate that if a station receives a channel management parameter containing a transmission definition index (txd) that it does not recognize, it will transmit a basic sync burst. | | | | | | |
| 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_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 100; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 15; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 0; txd:= 4) | Cb | Send a channel management message for directed slots with txd = 4 (not defined). |
| | 3 | record | RF | ctime:= time when the station under test receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 4 | await | | time:= ctime + (100 + 15) x 60/M1 | | Wait for the first sync burst from the station under test. |
| | 5 | verify | RF | SYNC_BASIC_a (s = add_A) transmitted in slot beginning at time:= ctime + (100 + 15) x 60/M1 | SBa | Verify that a basic sync burst is transmitted by the station under test at the requested time. |
| | 6 | rep 4 | | n:= 1 | | Start loop. |
| | 7 | await | | time:= ctime + (100 + 15) x 60/M1 + 60 x n | | Wait for a sync burst from the station under test. |

| Test Case Name: Channel_Unrecognize | | | | | | |
|---|------|---------|-----|---|-----|--|
| Purpose: To demonstrate that if a station receives a channel management parameter containing a transmission definition index (txd) that it does not recognize, it will transmit a basic sync burst. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 8 | verify | RF | SYNC_BASIC_a (s = add_A) transmitted in slot beginning at time:= ctime + (100 + 15) × 60/M1 + 60 × n | SBa | Verify that a basic sync burst is transmitted by the station under test at the requested time. |
| | 9 | end rep | | n:= n + 1 | | End loop. |
| postamble | 10 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_User_Trans | | | | | | |
|---|------|--------|-----|---|--------------------|--|
| Purpose: To demonstrate that a station will correctly interpret a channel management parameter with user-defined transmission blocks. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 4 | RF | id_val:= {0hex, 1hex, 9hex, 2hex}, aux_val:= {0, 1, 2, 0}, bg_val:= {0, 1, 0, 1} | | Repeat for different values of id_val, aux_val, bg_val. |
| | 3 | send | RF | CTRL_CHAN_h (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 0; csid:= 1; tc:= 2; fc:= 1; styp:= 0; sr:= 2; pc:= 0; sc:= 0; TRANS: id(1):= id_val; aux(1):= 0; b/g(1):= bg_val; id(2):= 1hex; aux(2):= 0; b/g(2):= 0; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq(1):= 0; nsd(1):= 1; txd(1):= 0; frq(2):= 0; nsd(2):= 1; txd(2):=1) | Ch | Send a channel management message from a simulated ground station G to the station under test with dc = 0, requesting transmission of a sync burst 2 times per minute for 5 minutes. |
| | 4 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 5 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 6 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 7 | verify | RF | SYNC BURST (s = add_A) transmitted before time = ctime + 60 | | Verify that the first sync burst transmitted in accordance with the channel management command is transmitted within 4 500 slots. |
| | 8 | verify | RF | IF (id_val, aux_val, bg_val) = (0hex, 0, 0) THEN SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 IF (id_val, aux_val, bg_val) = (1hex, 1, 1) THEN SYNC BURST = SYNC_HIGH_DYNAMIC_a and b/g = 1, br/gr = 1 IF (id_val, aux_val, bg_val) = (9hex, 2, 0) THEN SYNC BURST = SYNC_SINGLE_SLOT_TCP_a and b/g = 0, no = 2 IF (id_val, aux_val, bg_val) = (2hex, 2, 0) THEN SYNC BURST = SYNC_FULL_POSITION_a and b/g = 1 | SBa, SHa, SSa, SPa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |

| Test Case Name: | | Channel_User_Trans | | | | |
|------------------|------|--|-----|---|--------------------|---|
| Purpose: | | To demonstrate that a station will correctly interpret a channel management parameter with user-defined transmission blocks. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 9 | await | | time = sync_time + 30 | | Wait for the second sync burst from the station under test. |
| | 10 | verify | RF | SYNC_HIGH_DYNAMIC_a (s = add_A) transmitted in slot beginning at time = sync_time + 30 | | Verify that the second burst is of the correct type. |
| | 11 | verify | RF | SYNC_HIGH_DYNAMIC_a (s = add_A) contains b/g = 0, br/gr = 0 | | Verify that b/g and br/gr are set correctly. |
| | 12 | rep 8 | | n:= 1 | | Start loop. |
| | 13 | await | RF | SYNC BURST (s = add_A) in slot beginning at time = sync_time + n x 60 | | Wait for a sync burst from the station under test. |
| | 14 | verify | RF | IF (id_val, aux_val, bg_val) = (0hex, 0, 0) THEN SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 IF (id_val, aux_val, bg_val) = (1hex, 1, 1) THEN SYNC BURST = SYNC_HIGH_DYNAMIC_a and b/g = 1, br/gr = 1 IF (id_val, aux_val, bg_val) = (9hex, 2, 0) THEN SYNC BURST = SYNC_SINGLE_SLOT_TCP_a and b/g = 0, no = 2 IF (id_val, aux_val, bg_val) = (2hex, 2, 0) THEN SYNC BURST = SYNC_FULL_POSITION_a and b/g = 1 | SBa, SHa, SSa, SPa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 15 | await | RF | SYNC BURST (s = add_A) in slot beginning at time = sync_time + n x 60 + 30 | | Wait for a sync burst from the station under test. |
| | 16 | verify | RF | SYNC BURST = SYNC_HIGH_DYNAMIC_a (s = add_A) contains b/g = 0, br/gr = 0 | | Verify that b/g and br/gr are set correctly. |
| | 17 | endrep | | n:= n + 1 | | End loop. |
| | 18 | endrep | | next id_val, aux_val, bg_val | | Go to next value of (id_val, aux_val, bg_val). |
| postamble | 19 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Non_Sync_Trans | | | | |
|-----------------|-----------|--|-----|--|----------|--|
| Purpose: | | To demonstrate that a station will correctly interpret a channel management parameter with non-sync burst transmission blocks. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 4 | RF | mid_val:= {00000101, 01000101} | | Repeat for different values of id_val, aux_val, bg_val. |
| | 3 | send | RF | CTRL_CHAN_i (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 0; csid:= 1; tc:= 2; fc:= 1; styp:= 0; sr:= 2; pc:= 0; sc:= 0; TRANS: id:= 0; aux:= 0; b/g:= 0; mid:= mid_val; len:= 0; app:= 0; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq(1):= 0; nsd(1):= 1; txd(1):= 0; frq(2):= 0; nsd(2):= 1; txd(2):= 1) | Ci | Send a channel management message from a simulated ground station G to the station under test with dc = 0, requesting transmission of a sync burst 2 times per minute for 5 minutes. |
| | 4 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 5 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 6 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 7 | verify | RF | SYNC BURST (s = add_A) transmitted before time = ctime + 60 | | Verify that the first sync burst transmitted in accordance with the channel management command is transmitted within 4 500 slots. |
| | 8 | verify | RF | SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 | SBa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 9 | await | | time = sync_time + 30 | | Wait for the second burst from the station under test. |
| | 10 | verify | RF | IF mid_val = 00000101 THEN BURST = NO_OPERATION_a (s = add_A) IF mid_val = 01000101 THEN BURST = NET_ENTRY_a (s = add_A) transmitted in slot beginning at time = sync_time + 30 | NOa, NEa | Verify that the second burst is of the correct type. |
| | 11 | rep 8 | | n:= 1 | | Start loop. |
| | 12 | await | RF | SYNC BURST (s = add_A) in slot beginning at time = sync_time + n x 60 | | Wait for a sync burst from the station under test. |
| | 13 | verify | RF | SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 | SBa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 14 | await | RF | BURST (s = add_A) in slot beginning at time = sync_time + n x 60 + 30 | | Wait for the second burst from the station under test. |
| | 15 | verify | RF | IF mid_val = 00000101 THEN BURST = NO_OPERATION_a (s = add_A) IF mid_val = 01000101 THEN BURST = NET_ENTRY_a (s = add_A) transmitted in slot beginning at time = sync_time + n x 60 + 30 | NOa, NEa | Verify that the second burst is of the correct type. |
| | 16 | endrep | | n:= n + 1 | | End loop. |
| | 17 | endrep | | next mid_val | | Go to next value of mid_val. |
| | postamble | 18 | | | | |

Comments:

| Test Case Name: | | Channel_Speed | | | | |
|-----------------|------|---|-----|---|-----|--|
| Purpose: | | To demonstrate that if a mobile is travelling greater than 3 069 knots, then it will transmit a high dynamic variable field whenever it otherwise would have transmitted a basic, full position, or high resolution variable field. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | Velocity | | Input velocity as vel:= 3 073 knots | | Input velocity to the station under test that is above 3 069 knots. |
| | 3 | rep 4 | RF | id_val:= {0hex, 1hex, 9hex, 2hex}, aux_val:= {0, 1, 2, 0}, bg_val:= {0, 1, 0, 1} | | Repeat for different values of id_val, aux_val, bg_val. |
| | 4 | send | RF | CTRL_CHAN_h (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 0; csid:= 1; tc:= 2; fc:= 1; styp:= 0; sr:= 2; pc:= 0; sc:= 0; TRANS: id(1):= id_val; aux(1):= 0; b/g(1):= bg_val; id(2):= 1hex; aux(2):= 0; b/g(2):= 0; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq(1):= 0; nsd(1):= 1; txd(1):= 0; frq(2):= 0; nsd(2):= 1; txd(2):=1) | Ch | Send a channel management message from a simulated ground station G to the station under test with dc = 0, requesting transmission of a sync burst 2 times per minute for 5 minutes. |
| | 5 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 6 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 7 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 8 | verify | RF | SYNC BURST (s = add_A) transmitted before time = ctime + 60 | | Verify that the first sync burst transmitted in accordance with the channel management command is transmitted within 4 500 slots. |
| | 9 | verify | RF | IF (id_val, aux_val, bg_val) = (0hex, 0, 0) THEN SYNC BURST = SYNC_HIGH_DYNAMIC_a and b/g = 0, br/gr = 0 IF (id_val, aux_val, bg_val) = (1hex, 1, 1) THEN SYNC BURST = SYNC_HIGH_DYNAMIC_a and b/g = 1, br/gr = 1 IF (id_val, aux_val, bg_val) = (9hex, 2, 0) THEN SYNC BURST = SYNC_HIGH_DYNAMIC_a and b/g = 0, br/gr = 0 or 1 IF (id_val, aux_val, bg_val) = (2hex, 2, 0) THEN SYNC BURST = SYNC_HIGH_DYNAMIC_a and b/g = 1, br/gr = 0 | SHa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 10 | await | | time = sync_time + 30 | | Wait for the second sync burst from the station under test. |
| | 11 | verify | RF | SYNC_HIGH_DYNAMIC_a (s = add_A) transmitted in slot beginning at time = sync_time + 30 | | Verify that the second burst is of the correct type. |
| | 12 | verify | RF | SYNC_HIGH_DYNAMIC_a (s = add_A) contains b/g = 0, br/gr = 0 | | Verify that b/g and br/gr are set correctly. |
| | 13 | rep 8 | | n:= 1 | | Start loop. |

| Test Case Name: | | Channel_Speed | | | | |
|------------------|------|---|-----|---|--------------------|---|
| Purpose: | | To demonstrate that if a mobile is travelling greater than 3 069 knots, then it will transmit a high dynamic variable field whenever it otherwise would have transmitted a basic, full position, or high resolution variable field. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 14 | await | RF | SYNC BURST (s = add_A) in slot beginning at time = sync_time + n x 60 | | Wait for a sync burst from the station under test. |
| | 15 | verify | RF | IF (id_val, aux_val, bg_val) = (0hex, 0, 0) THEN SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 IF (id_val, aux_val, bg_val) = (1hex, 1, 1) THEN SYNC BURST = SYNC_HIGH_DYNAMIC_a and b/g = 1, br/gr = 1 IF (id_val, aux_val, bg_val) = (9hex, 2, 0) THEN SYNC BURST = SYNC_SINGLE_SLOT_TCP_a and b/g = 0, no = 2 IF (id_val, aux_val, bg_val) = (2hex, 2, 0) THEN SYNC BURST = SYNC_FULL_POSITION_a and b/g = 1 | SBa, SHa, SSa, SPa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 16 | await | RF | SYNC BURST (s = add_A) in slot beginning at time = sync_time + n x 60 + 30 | | Wait for a sync burst from the station under test. |
| | 17 | verify | RF | SYNC BURST = SYNC_HIGH_DYNAMIC_a (s = add_A) contains b/g = 0, br/gr = 0 | | Verify that b/g and br/gr are set correctly. |
| | 18 | endrep | | n:= n + 1 | | End loop. |
| | 19 | endrep | | next id_val, aux_val, bg_val | | Go to next value of (id_val, aux_val, bg_val). |
| postamble | 20 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_NotSupported | | | | |
|-----------------|------|---|-----|--|-----|--|
| Purpose: | | To demonstrate that whenever a mobile cannot transmit the information requested by a channel management CTRL, then the mobile will transmit a sync burst with a basic variable field. | | | | |
| 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_CHAN_j (s:= add_G; HEADER: dt:= 5; do:= 100; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 15; d:= add_A; TRANS: id:= 6hex; aux:= 0; b/g:= 0; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: frq:= 0; nsd:= 1; txd:= 0) | Cj | Send a channel management message for directed slots with txd = 4 (not defined). |
| | 3 | record | RF | ctime:= time when the station under test receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 4 | await | | time:= ctime + (100 + 15) x 60/M1 | | Wait for the first sync burst from the station under test. |

| Test Case Name: Channel_NotSupported | | | | | | |
|--|------|---------|-----|---|-----|--|
| Purpose: To demonstrate that whenever a mobile cannot transmit the information requested by a channel management CTRL, then the mobile will transmit a sync burst with a basic variable field. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 5 | verify | RF | SYNC_BASIC_a (s = add_A) transmitted in slot beginning at time:= ctime + (100 + 15) × 60/M1 | SBa | Verify that a basic sync burst is transmitted by the station under test at the requested time. |
| | 6 | rep 4 | | n:= 1 | | Start loop. |
| | 7 | await | | time:= ctime + (100 + 15) × 60/M1 + 60 × n | | Wait for a sync burst from the station under test. |
| | 8 | verify | RF | SYNC_BASIC_a (s = add_A) transmitted in slot beginning at time:= ctime + (100 + 15) × 60/M1 + 60 × n | SBa | Verify that a basic sync burst is transmitted by the station under test at the requested time. |
| | 9 | end rep | | n:= n + 1 | | End loop. |
| postamble | 10 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Script_Repeat | | | | | | |
|---|------|--------|-----|--|----------|--|
| Purpose: To demonstrate that a station will correctly interpret a channel management parameter with repeat count in the script block. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 2 | RF | id_val:= {0hex, 1hex} | | Repeat for different values of id_val. |
| | 3 | send | RF | CTRL_CHAN_k (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 0; csid:= 1; tc:= 2; fc:= 1; styp:= 0; sr:= 3; pc:= 0; sc:= 0; TRANS: id:= id_val; aux:= 0; b/g:= 0; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: rpt:= 0; frq:= 0; nsd:= 1; txd:= 0) | Ck | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst 3 times per minute for 5 minutes, with rpt:= 0. |
| | 4 | do | | ctime:= time when station A receives the transmission | | Record the time that the station under test receives the CTRL command. |
| | 5 | await | | SYNC BURST (s = add_A) | | Wait for a sync burst from the station under test. |
| | 6 | record | RF | sync_time:= time at beginning of slot containing SYNC BURST (s = add_A) | | Record the time the sync burst was transmitted. |
| | 7 | verify | RF | SYNC BURST (s = add_A) transmitted before time = ctime + 60 | | Verify that the first sync burst transmitted in accordance with the channel management command is transmitted within 4 500 slots. |
| | 8 | verify | RF | IF (id_val, aux_val, bg_val) = (0hex, 0, 0) THEN SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 IF (id_val, aux_val, bg_val) = (1hex, 1, 1) THEN SYNC BURST = SYNC_HIGH_DYNAMIC_a and b/g = 0, br/gr = 0 | SBa, SHa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 9 | rep 14 | | n:= 1 | | Start loop. |

| Test Case Name: | | Channel_Script_Repeat | | | | |
|------------------|------|--|-----|---|----------|---|
| Purpose: | | To demonstrate that a station will correctly interpret a channel management parameter with repeat count in the script block. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 10 | await | RF | SYNC BURST (s = add_A) in slot beginning at time = sync_time + n × 20 | | Wait for a sync burst from the station under test. |
| | 11 | verify | RF | IF (id_val, aux_val, bg_val) = (0hex, 0, 0) THEN SYNC BURST = SYNC_BASIC_a and b/g = 0, br/gr = 0 IF (id_val, aux_val, bg_val) = (1hex, 1, 1) THEN SYNC BURST = SYNC_HIGH_DYNAMIC_a and b/g = 0, br/gr = 0 | SBa, SHa | Verify that the sync burst is of the expected type and has the b/g and br/gr flags set appropriately. |
| | 12 | endrep | | n:= n + 1 | | End loop. |
| | 13 | endrep | | next id_val | | Go to next value of id_val. |
| postamble | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Discard | | | | |
|------------------|------|---|-----|---|-----|---|
| Purpose: | | To demonstrate that a receiving station will discard a channel management CTRL command containing a script block with two consecutive octets having bits 6 to 8 equal to "111". | | | | |
| 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_CHAN_I (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 0; csid:= 1; tc:= 2; fc:= 1; styp:= 0; sr:= 3; pc:= 0; sc:= 0; TRANS: id:= id_val; aux:= 0; b/g:= 0; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; SCRIPT: rpt:= 0; rpt:= 0; frq:= 0; nsd:= 1; txd:= 0) | Cl | Send a channel management message from a simulated ground station G to the station under test with a script block with two octets with frq="111" plus a third "normal" octet. |
| | 3 | rep 60 | | n:= 1 | | Start loop. |
| | 4 | await | RF | SYNC BURST (s = add_A) | Sb | Wait for a sync burst from the station under test. |
| | 5 | record | RF | time(n):= time at the beginning of slot containing SYNC BURST (s = add_A) | | Record the time that the message was received. |
| | 6 | endrep | | n:= n + 1 | | End loop. |
| | 7 | verify | | time(60) - time(1) = 5 mins | | Verify that the station under test does not transmit according to the instructions of the channel management parameter, but continues autonomous behaviour on the GSCs (i.e. 12 messages in 1 min). |
| postamble | 8 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Param_Q1 | | | | |
|-----------------|------|---|-----|---|-----|--|
| Purpose: | | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the Q1 parameter. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | rep 2 | | priority1:= {11, 13}, txd1:= {0, 2} priority2:= {13, 11}, txd2:= {2, 0} | | Repeat the test with the higher priority CTRL coming first. Verify that the result is the same. |
| test body | 3 | do | | M_SYNC_BURST_SEND (2 249, 140 NM) | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 249 slots. The same slots are reserved for another 3 superframes. |
| | 4 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |
| | 5 | do | | M_SYNC_BURST_SEND (2 246, 140 NM) beginning at time:= send_time + 2 250 x 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 246 slots. The same slots are reserved for another 3 superframes. |
| | 6 | await | | time:= send_time + 4 496 x 60/M1 | | Wait for the first available slot. |
| | 7 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: Q1:= priority1; Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; V12, TV11min, TV11max, V11, VS2, VS4 set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 496 x 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with Q1 = priority1. |
| | 8 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: Q1:= priority2; Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; Q4, V12, TV11min, TV11max, V11, VS2, VS4 set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd2) in slot beginning at time:= send_time + 4 498 x 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with Q1 = priority2. |
| | 9 | await | | time:= send_time + 60 + 2 249 x 60/M1 | | Wait for the next available slot. |
| | 10 | verify | RF | SYNC_FULL_POSITION_a (s = add_A) transmitted in slot beginning at time:= send_time + 60 + 2 249 x 60/M1 | | Verify that the station under test transmits the sync burst with full position variable part, and therefore with the higher priority. |
| | 11 | await | | time:= send_time + 120 + 2 249 x 60/M1 | | Wait for the next available slot. |
| | 12 | verify | RF | SYNC_FULL_POSITION_a (s = add_A) transmitted in slot beginning at time:= send_time + 120 + 2 249 x 60/M1 | | Verify that the station under test transmits the sync burst with full position variable part, and therefore with the higher priority. |
| | 13 | end rep | | next priority1, priority2, txd1, txd2 | | Select the next values and repeat. |

| Test Case Name: Channel_Param_Q1 | | | | | | |
|--|------|--------|-----|------------------|-----|---------------------------------------|
| Purpose: To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the Q1 parameter. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| postamble | 14 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Param_Q4 | | | | | | |
|--|------|--------|-----|--|-----|--|
| Purpose: To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the Q4 parameter. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_SYNC_BURST_SEND (2 249, 140 NM) | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 249 slots. The same slots are reserved for another 3 superframes. |
| | 3 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |
| | 4 | do | | M_SYNC_BURST_SEND (2 248, 140 NM) beginning at time:= send_time + 2 250 × 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 248 slots. The same slots are reserved for another 3 superframes. |
| | 5 | await | | time:= send_time + 4 498 × 60/M1 | | Wait for the first available slot. |
| | 6 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: Q4:= 5; Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; Q1, V12, TV11min, TV11max, V11, VS2, VS4 set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 498 × 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with Q4 = 5. |
| | 7 | await | | time:= send_time + 60 + 2 249 × 60/M1 | | Wait for the next available slot. |
| | 8 | verify | RF | SYNC_FULL_POSITION_a (s = add_A) not transmitted in slot beginning at time:= send_time + 60 + 2 249 × 60/M1 | | Verify that the station under test is unable to transmit the sync burst with full position variable part, and therefore is applying the Q4 parameter. |
| | 9 | wait | | 4 minutes | | Wait until all the reservations have expired. |
| | 10 | do | | M_SYNC_BURST_SEND (2 249, 140 NM) beginning at time:= send_time | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 249 slots. The same slots are reserved for another 3 superframes. |
| | 11 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |

| Test Case Name: Channel_Param_Q4 | | | | | | |
|--|------|--------|-----|---|-----|--|
| Purpose: To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the Q4 parameter. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 12 | do | | M_SYNC_BURST_SEND (2 248, 140 NM) beginning at time:= send_time + 2 250 × 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 248 slots. The same slots are reserved for another 3 superframes. |
| | 13 | await | | time:= send_time + 4 498 × 60/M1 | | Wait for the first available slot. |
| | 14 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: Q4:= 5; Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; Q1, V12, TV11min, TV11max, V11, VS2, VS4 set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 498 × 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with Q4 = 1. |
| | 15 | await | | time:= send_time + 60 + 2 249 × 60/M1 | | Wait for the next available slot. |
| | 16 | verify | RF | SYNC_FULL_POSITION_a (s = add_A) transmitted in slot beginning at time:= send_time + 60 + 2 249 × 60/M1 | | Verify that the station under test is able to transmit the sync burst with full position variable part, and therefore is applying the Q4 parameter. |
| postamble | 17 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Param_Q2 | | | | | | |
|---|------|--------|-----|--|-----|---|
| Purpose: To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the Q2 parameters. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_SYNC_BURST_SEND (2 249, 140 NM) | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 249 slots. The same slots are reserved for another 3 superframes. |
| | 3 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |
| | 4 | do | | M_SYNC_BURST_SEND (2 248, 140 NM) beginning at time:= send_time + 2 250 × 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 248 slots. The same slots are reserved for another 3 superframes. |
| | 5 | await | | time:= send_time + 4 498 × 60/M1 | | Wait for the first available slot. |

| Test Case Name: | | Channel_Param_Q2 | | | | |
|------------------|------|--|-----|--|-----|---|
| Purpose: | | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the Q2 parameters. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 6 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; Q4, Q1, V12, TV11min, TV11max, V11, VS2, VS4 set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 498 × 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with all Q2 parameters set to 150 NM. |
| | 7 | await | | time:= send_time + 60 + 2 249 × 60/M1 | | Wait for the next available slot. |
| | 8 | verify | RF | SYNC_FULL_POSITION_a (s = add_A) not transmitted in slot beginning at time:= send_time + 60 + 2 249 × 60/M1 | | Verify that the station under test is unable to transmit the sync burst with full position variable part, and therefore is applying the Q2 parameter. |
| | 9 | wait | | 4 minutes | | Wait until all the reservations have expired. |
| | 10 | do | | M_SYNC_BURST_SEND (2 249, 140 NM) beginning at time:= send_time | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 249 slots. The same slots are reserved for another 3 superframes. |
| | 11 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |
| | 12 | do | | M_SYNC_BURST_SEND (2 248, 140 NM) beginning at time:= send_time + 2 250 × 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 248 slots. The same slots are reserved for another 3 superframes. |
| | 13 | await | | time:= send_time + 4 498 × 60/M1 | | Wait for the first available slot. |
| | 14 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: Q2a = 130 NM; Q2b = 130 NM; Q2c = 130 NM; Q2d = 130 NM; Q4, Q1, V12, TV11min, TV11max, V11, VS2, VS4 set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 498 × 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with all Q2 parameters set to 130 NM. |
| | 15 | await | | time:= send_time + 60 + 2 249 × 60/M1 | | Wait for the next available slot. |
| | 16 | verify | RF | SYNC_FULL_POSITION_a (s = add_A) transmitted in slot beginning at time:= send_time + 60 + 2 249 × 60/M1 | | Verify that the station under test is able to transmit the sync burst with full position variable part, and therefore is applying the Q2 parameters. |
| postamble | 17 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Param_V12 | | | | |
|-----------------|------|--|-----|--|-----|---|
| Purpose: | | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the V12 parameter. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_SYNC_BURST_SEND (2 245, 140 NM) | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 245 slots. The same slots are reserved for another 3 superframes. |
| | 3 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |
| | 4 | await | | time:= send_time + 2 247 × 60/M1 | | |
| | 5 | send | RF | SYNC_BURST_b (s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM); pt:= 3; po:= 0) in slot beginning at time:= send_time + 2 247 × 60/M1 | Sb | Send a one-slot sync burst from a simulated station B which is < Q2a,b,c,d from the station under test. This sync burst reserves 1 slot with two free slots either side. |
| | 6 | do | | M_SYNC_BURST_SEND (2 248, 140 NM) beginning at time:= send_time + 2 250 × 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 248 slots. The same slots are reserved for another 3 superframes. |
| | 7 | await | | time:= send_time + 4 498 × 60/M1 | | Wait for the first available slot. |
| | 8 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: Q4:= 3; V12:= (2/M1) × V11; Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; Q1, TV11min, TV11max, V11, VS2, VS4 set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 498 × 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with Q4 = 3 and V12 = (2/M1)×V11 (3 slots). |
| | 9 | rep 4 | | n:= {0, 1, 3, 4}; q:= FALSE | | Repeat over the four available slots. |
| | 10 | await | | time:= send_time + 60 + (2 245 + n) × 60/M1 | | Wait for the next available slot. |
| | 11 | verify | RF | IF SYNC_FULL_POSITION_a (s = add_A) transmitted in slot beginning at time:= send_time + 60 + (2 245 + n) × 60/M1 THEN q:= TRUE | | Verify that the station under test is able to transmit the sync burst with full position variable part, and therefore is applying the V12 parameter. |
| | 12 | endrep | | next n | | |
| | 13 | verify | | q:= FALSE | | Verify that q = FALSE, that the station under test is unable to transmit the sync burst with full position variable part, and therefore is applying the V12 parameter. |
| | 14 | wait | | 4 minutes | | Wait until all the reservations have expired. |
| | 15 | do | | M_SYNC_BURST_SEND (2 245, 140 NM) beginning at time:= send_time | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 249 slots. The same slots are reserved for another 3 superframes. |
| | 16 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |

| Test Case Name: | | Channel_Param_V12 | | | | |
|------------------|------|--|-----|--|-----|---|
| Purpose: | | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the V12 parameter. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 17 | await | | time:= send_time + 2 247 × 60/M1 | | |
| | 18 | send | RF | SYNC_BURST_b (s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM); pt:= 3; po:= 0) in slot beginning at time:= send_time + 2 247 × 60/M1 | Sb | Send a one-slot sync burst from station B which is < Q2a,b,c,d from the station under test. This sync burst reserves 1 slot with two free slots either side. |
| | 19 | do | | M_SYNC_BURST_SEND (2 248, 140 NM) beginning at time:= send_time + 2 250 × 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 248 slots. The same slots are reserved for another 3 superframes. |
| | 20 | await | | time:= send_time + 4 498 × 60/M1 | | Wait for the first available slot. |
| | 21 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: Q4:= 3; V12:= (4/M1) × V11; Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; Q1, TV11min, TV11max, V11, VS2, VS4 set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 498 × 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with Q4 = 3 and V12 = (4/M1)×V11 (5 slots). |
| | 22 | rep 4 | | n:= {0, 1, 3, 4}; q:= FALSE | | Repeat over the four available slots. |
| | 23 | await | | time:= send_time + 60 + (2 245 + n) × 60/M1 | | Wait for the next available slot. |
| | 24 | verify | RF | IF SYNC_FULL_POSITION_a (s = add_A) transmitted in slot beginning at time:= send_time + 60 + (2 245 + n) × 60/M1 THEN q:= TRUE | | Verify that the station under test is able to transmit the sync burst with full position variable part, and therefore is applying the V12 parameter. |
| | 25 | endrep | | next n | | |
| | 26 | verify | | q:= TRUE | | Verify that q = TRUE, that the station under test is able to transmit the sync burst with full position variable part, and therefore is applying the V12 parameter. |
| postamble | 27 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | Channel_Param_VS4 | | | | |
|-----------------|------|--|-----|---|-----|--|
| Purpose: | | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the VS4 parameter. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | do | | M_SYNC_BURST_SEND (2 248, 140 NM) | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 248 slots. The same slots are reserved for another 3 superframes. |
| | 3 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |
| | 4 | await | | time:= send_time + 2 248 × 60/M1 | | |
| | 5 | send | RF | SYNC_BURST_b (s:= add_G; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM); pt:= 3; po:= 0) in slot beginning at time:= send_time + 2 247 × 60/M1 | Sb | Send a one slot sync burst from a simulated ground station G which is 140 NM away and < Q2a,b,c,d from the station under test. This sync burst reserves 4 quarantine slots after it. |
| | 6 | do | | M_SYNC_BURST_SEND (2 248, 140 NM) beginning at time:= send_time + 2 250 × 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 248 slots. The same slots are reserved for another 3 superframes. |
| | 7 | await | | time:= send_time + 4 498 × 60/M1 | | Wait for the first available slot. |
| | 8 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: VS4:= 160 NM; Q4:= 1; Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; Q1, V12, TV11min, TV11max, V11, VS2, set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 498 × 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with VS4 = 160 NM. |
| | 9 | await | | time:= send_time + 60 + 2 249 × 60/M1 | | Wait for the next available slot. |
| | 10 | verify | RF | SYNC_FULL_POSITION_a (s = add_A) not transmitted in slot beginning at time:= send_time + 60 + 2 249 × 60/M1 | | Verify that the station under test is unable to transmit the sync burst with full position variable part, and therefore is applying the VS4 parameter. |
| | 11 | wait | | 4 minutes | | Wait until all the reservations have expired. |
| | 12 | do | | M_SYNC_BURST_SEND (2 248, 140 NM) | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 248 slots. The same slots are reserved for another 3 superframes. |
| | 13 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |
| | 14 | await | | time:= send_time + 2 248 × 60/M1 | | |
| | 15 | send | RF | SYNC_BURST_b (s:= add_G; lat:= CPR_LAT(0); lon:= CPR_LON(E 9 NM); pt:= 3; po:= 0) in slot beginning at time:= send_time + 2 247 × 60/M1 | Sb | Send a one slot sync burst from a simulated ground station G which is 140 NM away and < Q2a,b,c,d from the station under test. This sync burst reserves 4 quarantine slots after it. |

| Test Case Name: Channel_Param_VS4 | | | | | | |
|---|------|--------|-----|---|-----|--|
| Purpose: To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the VS4 parameter. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 16 | do | | M_SYNC_BURST_SEND (2 248, 140 NM) beginning at time:= send_time + 2 250 × 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 248 slots. The same slots are reserved for another 3 superframes. |
| | 17 | await | | time:= send_time + 4 498 × 60/M1 | | Wait for the first available slot. |
| | 18 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: VS4:= 120 NM; Q4:= 1; Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; Q1, V12, TV11min, TV11max, V11, VS2, set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 498 × 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with VS4 = 120 NM. |
| | 19 | await | | time:= send_time + 60 + 2 249 × 60/M1 | | Wait for the next available slot. |
| | 20 | verify | RF | SYNC_FULL_POSITION_a (s = add_A) transmitted in slot beginning at time:= send_time + 60 + 2 249 × 60/M1 | | Verify that the station under test is able to transmit the sync burst with full position variable part, and therefore is applying the VS4 parameter. |
| postamble | 21 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: Channel_Param_VS2 | | | | | | |
|---|------|--------|-----|---|-----|---|
| Purpose: To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the VS2 parameter. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | SYNC_BURST_b (s:= add_C; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM); pt:= 3; po:= 0) | Sb | Send a one slot sync burst from a simulated station C which is 140 NM away and < Q2a,b,c,d from the station under test. |
| | 3 | send | RF | SYNC_BURST_b (s:= add_D; lat:= CPR_LAT(N 31 NM); lon:= CPR_LON(E 140 NM); pt:= 3; po:= 0) | Sb | Send a one slot sync burst from a simulated station D which is 140 NM away and < Q2a,b,c,d from the station under test. D's position is such that a transmission from C to D is not CCI protected with a CCI ratio of 5 (VS2 = 14) but would be with a CCI ratio of 4 (VS2 = 12). |
| | 4 | do | | M_SYNC_BURST_SEND (2 249, 140 NM) | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 249 slots. The same slots are reserved for another 3 superframes. |
| | 5 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |

| Test Case Name: | | Channel_Param_VS2 | | | | |
|-----------------|------|--|-----|---|-----|---|
| Purpose: | | To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the VS2 parameter. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 6 | do | | M_SYNC_BURST_SEND (2 247, 140 NM) beginning at time:= send_time + 2 250 × 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 247 slots. The same slots are reserved for another 3 superframes. |
| | 7 | await | | time:= send_time + 4 497 × 60/M1 | | Wait for the first available slot. |
| | 8 | send | RF | ADSB_REQUEST_a (s:= add_C; d:=add_D; sdf:= 1; ro:= 2 252) in slot beginning at time:= send_time + 4 497 × 60/M1 | ARa | In slot 4 497, send a unicast burst from station C to station D reserving a slot for a point-to-point transmission from C to D in slot 6 749. |
| | 9 | await | | time:= send_time + 4 498 × 60/M1 | | Wait for the first available slot. |
| | 10 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: Q4:= 1; VS2:= 14 dB; Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; Q1, V12, TV11min, TV11max, V11, VS4 set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 498 × 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with VS2 = 14 dB. |
| | 11 | await | | time:= send_time + 60 + 2 249 × 60/M1 | | Wait for the next available slot. |
| | 12 | verify | RF | SYNC_FULL_POSITION_a (s = add_A) not transmitted in slot beginning at time:= send_time + 60 + 2 249 × 60/M1 | | Verify that the station under test is unable to transmit the sync burst with full position variable part, and therefore is applying the VS2 parameter. |
| | 13 | wait | | 4 minutes | | Wait until all the reservations have expired. |
| | 14 | send | RF | SYNC_BURST_b (s:= add_C; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM); pt:= 3; po:= 0) | Sb | Send a one slot sync burst from a simulated station C which is 140 NM away and < Q2a,b,c,d from the station under test. |
| | 15 | send | RF | SYNC_BURST_b (s:= add_D; lat:= CPR_LAT(N 31 NM); lon:= CPR_LON(E 140 NM); pt:= 3; po:= 0) | Sb | Send a one slot sync burst from a simulated station D which is 140 NM away and < Q2a,b,c,d from the station under test. D's position is such that a transmission from C to D is not CCI protected with a CCI ratio of 5 (VS2 = 14) but would be with a CCI ratio of 4 (VS2 = 12). |
| | 16 | do | | M_SYNC_BURST_SEND (2 249, 140 NM) | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 249 slots. The same slots are reserved for another 3 superframes. |
| | 17 | record | | send_time:= time at beginning of first sync burst sent by macro | | Record the time of the first sync burst sent by the macro. |
| | 18 | do | | M_SYNC_BURST_SEND (2 247, 140 NM) beginning at time:= send_time + 2 250 × 60/M1 | | Send sync bursts from other simulated stations at 140 NM range, occupying 2 247 slots. The same slots are reserved for another 3 superframes. |
| | 19 | await | | time:= send_time + 4 497 × 60/M1 | | Wait for the first available slot. |

| Test Case Name: Channel_Param_VS2 | | | | | | |
|---|------|--------|-----|---|-----|---|
| Purpose: To demonstrate that a station in receipt of a channel management CTRL will transmit all of the streams defined in the script while applying the VS2 parameter. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 20 | send | RF | ADSB_REQUEST_a (s:= add_C; d:=add_D; sdf:= 1; ro:= 2 252) in slot beginning at time:= send_time + 4 497 × 60/M1 | ARa | In slot 4 497, send a unicast burst from station C to station D reserving a slot for a point-to-point transmission from C to D in slot 6 749. |
| | 21 | await | | time:= send_time + 4 498 × 60/M1 | | Wait for the first available slot. |
| | 22 | send | RF | CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 1; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 0; d:= add_A; FREQ: sleep:= 0; auto:= 0; rcv:= 0; f:= transmit freq; PARAM: Q4:= 1; VS2:= 12 dB; Q2a = 150 NM; Q2b = 150 NM; Q2c = 150 NM; Q2d = 150 NM; Q1, V12, TV11min, TV11max, V11, VS2, VS4 set to default; SCRIPT: frq:= 0; nsd:= 0; txd:= txd1) in slot beginning at time:= send_time + 4 498 × 60/M1 | Cm | Send a channel management message from a simulated ground station G to the station under test, requesting transmission of a sync burst once per minute for 4 minutes, with VS2 = 12 dB. |
| | 23 | await | | time:= send_time + 60 + 2 249 × 60/M1 | | Wait for the next available slot. |
| | 24 | verify | RF | SYNC_FULL_POSITION_a (s = add_A) transmitted in slot beginning at time:= send_time + 60 + 2 249 × 60/M1 | | Verify that the station under test is able to transmit the sync burst with full position variable part, and therefore is applying the Q4 parameter. |
| postamble | 25 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: CPR_Encode_Offset | | | | | | |
|--|------|----------|----------|--|----------|---|
| Purpose: To demonstrate that offset latitude and longitude positions are correctly encoded in the sync burst information fields using the CPR algorithm. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | send | VSS | SUPPRESS AUTONOMOUS SYNC BURSTS | | Suppress the autonomous sync bursts to avoid possible confliction. |
| | 3 | send | VSS | SET PARAMETERS (p:= 1) | | Ensure 100 % chance of transmission on access to hasten sync burst responses following a general request. |
| test body | 4 | rep 2166 | | n:= 1; initialize p | | |
| | 5 | send | Position | Input to station under test: LAT(n):= 12,8557 + n × 0,163 LON(n):= -0,8150 + n × 0,163 | CE(r, c) | Send test values of latitude and longitude from CPR_ENC_TABLE to the station under test. |
| | 6 | rep 135 | | k:= 1 | | |

| Test Case Name: | | CPR_Encode_Offset | | | | |
|--|------|-------------------|-----|--|-----------------|---|
| Purpose: To demonstrate that offset latitude and longitude positions are correctly encoded in the sync burst information fields using the CPR algorithm. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 7 | do | | IF LAT(n) = CPR_ENC_TABLE (k, latitude) for row k of table AND LON(n) = CPR_ENC_TABLE (k, longitude) for row k of table THEN p:= k continue with following test steps within loop using current p value ELSE go to next n bypassing all the steps before the end of the loop | | |
| | 8 | endrep | | k:= k + 1 | | |
| | 9 | send | RF | ADSB_REQUEST_a (r-id:= 0 hex; s= add_B) | ARa | Send a general request burst from a simulated station B, requesting the station under test to transmit a sync burst with a basic variable information field. |
| | 10 | await | RF | SYNC_BURST_g (s= add_A) | Sg | |
| | 11 | do | | IF cprf in fixed part of SYNC_BURST_g (s= add_A) equals 0 THEN continue with following test steps within n loop ELSE exit n loop and start n loop again with n:=1 | | Restart n loop if for the first pair of latitude and longitude values which coincides with those in the first row of CPR_ENC_TABLE, the CPR type cprf is not zero. NOTE: The test values provided in the CPR_ENC_TABLE can only be used if the CPR type happens to correspond to the type for which the test values were calculated. If this is not the case when the n test loop starts for the first time, the n test loop must be restarted until this happens. |
| | 12 | verify | RF | In fixed part of SYNC_BURST_g (s= add_A): cprf = CPR_ENC_TABLE (p, cpr_type) | Sg, CE(r, c) | Verify that the encoded values of latitude, longitude, and CPR type in the sync burst from the station under test agree with the values given in CPR_ENC_TABLE. |
| | | verify | RF | AND lat = CPR_ENC_TABLE (p, lat_enc) | | |
| | | verify | RF | AND lon = CPR_ENC_TABLE (p, lon_enc) | | |
| | 13 | verify | RF | In variable part of SYNC_BURST_g (s= add_A): pid = CPR_ENC_TABLE (p, pid_enc) | Sg, CE(r, c) | Verify that the encoded values of pid, lat6, and lon6 in the variable part of the sync burst from the station under test agree with the values given in CPR_ENC_TABLE. |
| | | verify | RF | AND lat6(bits 1-5) = CPR_ENC_TABLE (p, lat6(1-5)) | | |
| | | verify | RF | AND lat6(bit 6) = CPR_ENC_TABLE (p, lat6(6)) | | |
| | | verify | RF | AND lon6(bits 1-5) = CPR_ENC_TABLE (p, lon6(1-5)) | | |
| | | verify | RF | AND lon6(bit 6) = CPR_ENC_TABLE (p, lon6(6)) | | |

| Test Case Name: | | CPR_Encode_Offset | | | | |
|--|------|-------------------|-----|---|-----------------|--|
| Purpose: To demonstrate that offset latitude and longitude positions are correctly encoded in the sync burst information fields using the CPR algorithm. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 14 | send | RF | ADSB_REQUEST_a (r-id:= 1 hex; s= add_B) | ARa | Send a general request burst from a simulated station B, requesting the station under test to transmit a sync burst with a high dynamic variable information field. |
| | 15 | await | RF | SYNC_BURST_h (s= add_A) | Sh | |
| | 16 | verify | RF | In variable part of SYNC_BURST_h (s= add_A): lat4(bits 1-3) = CPR_ENC_TABLE (p, lat4(1-3)) AND | Sh, CE(r, c) | Verify that the encoded values of lat4 and lon4 in the variable part of the sync burst from the station under test agree with the values given in CPR_ENC_TABLE. |
| | | verify | RF | lat4(bit 4) = CPR_ENC_TABLE (p, lat4(4)) AND | | |
| | | verify | RF | lon4(bits 1-3) = CPR_ENC_TABLE (p, lon4(1-3)) AND | | |
| | | verify | RF | lon4(bit 4) = CPR_ENC_TABLE (p, lon4(4)) | | |
| | 17 | send | RF | ADSB_REQUEST_a (r-id:= 2 hex; s= add_B) | ARa | Send a general request burst from a simulated station B, requesting the station under test to transmit a sync burst with a full position variable information field. |
| | 18 | await | RF | SYNC_BURST_h (s= add_A) | Sh | |
| | 19 | verify | RF | In variable part of SYNC_BURST_h (s= add_A): pid = CPR_ENC_TABLE (p, pid_enc) | Sh, CE(r, c) | Verify that the encoded values of pid in the variable part of the sync burst from the station under test agree with the values given in CPR_ENC_TABLE. |
| | 20 | send | RF | ADSB_REQUEST_a (r-id:= AA0 hex; s= add_B) | ARa | Send a General Request burst from a simulated station B, requesting the station under test to transmit a sync burst with a high resolution variable information field. |
| | 21 | await | RF | SYNC_BURST_i (s= add_A) | Si | |
| | 22 | verify | RF | In variable part of SYNC_BURST_i (s= add_A): lat8(bits 1 to 7) = CPR_ENC_TABLE (p, lat8(1-7)) AND | Si, CE(r, c) | Verify that the encoded values of lat8 and lon8 in the variable part of the sync burst from the station under test agree with the values given in CPR_ENC_TABLE. |
| | | verify | RF | lat8(bit 8) = CPR_ENC_TABLE (p, lat8(8)) AND | | |
| | | verify | RF | lon8(bits 1 to 7) = CPR_ENC_TABLE (p, lon8(1-7)) AND | | |
| | | verify | RF | lon8(bit 8) = CPR_ENC_TABLE (p, lon8(8)) | | |
| | 23 | endrep | | n:= n + 1 | | |
| postamble | 24 | send | VSS | SET PARAMETERS (p:= 64/256) | | Restore to default value. |
| | 25 | send | VSS | REINSTATE AUTONOMOUS SYNC BURSTS | | Reinstate the autonomous sync bursts. |
| Comments: | | | | | | |

| Test Case Name: | | CPR_Decode_Offset | | | | |
|--|------|-------------------|--------|--|-----------------|---|
| Purpose: To demonstrate that offset latitude and longitude positions are correctly decoded from the sync burst information fields using the CPR algorithm. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| | 2 | send | VSS | SUPPRESS AUTONOMOUS SYNC BURSTS | | Suppress the autonomous sync bursts to avoid possible confliction. |
| | 3 | send | VSS | SET PARAMETERS (G1:= 10) | | Set the maximum number of missed reservations to 10. |
| test body | 4 | rep 135 | | n:= 1 | | |
| | 5 | send | RF | SYNC_BURST_g (po:= 0; pt:= 0; s:= add_B; lat:= CPR_ENC_TABLE (n, lat_enc); lon:= CPR_ENC_TABLE (n, lon_enc); lat6(bits 1-5):= CPR_ENC_TABLE (n, lat6(1-5)); lat6(bit 6):= CPR_ENC_TABLE (n, lat6(6)); lon6(bits 1-5):= CPR_ENC_TABLE (n, lon6(1-5)) lon6(bit 6):= CPR_ENC_TABLE (n, lon6(6))) | Sg, CE(r, c) | Send a sync burst from a simulated station B with a basic variable information field. The encoded values for lat and lon in the fixed part of the burst, and for lat6 and lon6 in the variable part, are taken from row n of CPR_ENC_TABLE. |
| | 6 | await | AppOut | LAT DATA OUT, LON DATA OUT | | Wait for the received sync burst to be processed by the station under test and sent to the ADS application output. |
| | 7 | verify | AppOut | LAT DATA OUT = CPR_DEC_TABLE (n, decoded 6-bit high-res lat) AND LON DATA OUT = CPR_DEC_TABLE (n, decoded 6-bit high-res lon) | CD(r, c) | Verify that the station under test processes the data and generates the appropriate output for display to the aircrew. |
| | 8 | send | RF | SYNC_BURST_h (po:= 0; pt:= 0; s:= add_B; pid:= CPR_ENC_TABLE (n, pid_enc); lat:= CPR_ENC_TABLE (n, lat_enc); lon:= CPR_ENC_TABLE (n, lon_enc); lat4(bits 1-3):= CPR_ENC_TABLE (n, lat4(1-3)); lat4(bit 4):= CPR_ENC_TABLE (n, lat4(4)); lon4(bits 1-3):= CPR_ENC_TABLE (n, lon4(1-3)) lon4(bit 4):= CPR_ENC_TABLE (n, lon4(4))) | Sh, CE(r, c) | Send a sync burst from a simulated station B with a basic ground variable information field. The encoded values for lat and lon in the fixed part of the burst, and for pid, lat4, and lon4 in the variable part, are taken from row n of CPR_ENC_TABLE. |
| | 9 | await | AppOut | LAT DATA OUT, LON DATA OUT | | Wait for the received sync burst to be processed by the station under test and sent to the ADS application output. |
| | 10 | verify | AppOut | LAT DATA OUT = CPR_DEC_TABLE (n, decoded 4-bit high-res lat) AND LON DATA OUT = CPR_DEC_TABLE (n, decoded 4-bit high-res lon) | CD(r, c) | Verify that the station under test processes the data and generates the appropriate output for display to the aircrew. |
| | 11 | send | RF | SYNC_BURST_i (po:= 0; pt:= 0; s:= add_B; lat:= CPR_ENC_TABLE (n, lat_enc); lon:= CPR_ENC_TABLE (n, lon_enc); lat8(bits 1 to 7):= CPR_ENC_TABLE (n, lat8(1-7)); lat8(bit 8):= CPR_ENC_TABLE (n, lat8(8)); lon8(bits 1 to 7):= CPR_ENC_TABLE (n, lon8(1-7)) lon8(bit 8):= CPR_ENC_TABLE (n, lon8(8))) | Si, CE(r, c) | Send a sync burst from a simulated station B with a basic variable information field. The encoded values for lat and lon in the fixed part of the burst, and for lat8 and lon8 in the variable part, are taken from row n of CPR_ENC_TABLE. |
| | 12 | await | AppOut | LAT DATA OUT, LON DATA OUT | | Wait for the received sync burst to be processed by the station under test and sent to the ADS application output. |

| Test Case Name: CPR_Decode_Offset | | | | | | |
|--|------|--------|--------|---|----------|--|
| Purpose: To demonstrate that offset latitude and longitude positions are correctly decoded from the sync burst information fields using the CPR algorithm. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 13 | verify | AppOut | LAT DATA OUT = CPR_DEC_TABLE (n, decoded 8-bit high-res lat) AND LON DATA OUT = CPR_DEC_TABLE (n, decoded 8-bit high-res lon) | CD(r, c) | Verify that the station under test processes the data and generates the appropriate output for display to the aircrew. |
| | 14 | endrep | | n:= n + 1 | | |
| postamble | 15 | send | VSS | SET PARAMETERS (G1:= 3) | | Restore to default value. |
| | 16 | send | VSS | REINSTATE AUTONOMOUS SYNC BURSTS | | Reinstate the autonomous sync bursts. |
| Comments: | | | | | | |

7.4.3.2.2.1

TIS-B test cases

| Test Case Name: TIS_TIV_Unique_A | | | | | | |
|---|------|--------|--------|--|-----|--|
| Purpose: To demonstrate that the same TIV ID in target messages referring to the same aircraft, but received from different ground stations, are identified as separate TIVs. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | TIS_MAN_a (s:= add_G; tivid:=0; nvert:= 0; lat:= CPR_LAT(N 50,5 NM); long:= CPR_LON(E 1,3 NM)) | TMa | Send a management message to define TIV 0 from simulated ground station G with centre at 50,5 NM North, 1,3 NM East. |
| | 3 | send | RF | TIS_AC_AIR_a (s:= add_G; tivid:=0; call:= BA1234; cat:= 2) | TAa | Send to the station under test an aircraft target (airborne) message from ground station G referring to TIV 0. |
| | 4 | send | RF | TIS_MAN_a (s:= add_H; tivid:=0; d:= add_A; nvert:= 0; lat:= CPR_LAT(N 52.3 NM); long:= CPR_LON(0)) | TMa | Send a management message to define TIV 0 from simulated ground station H with different centre coordinates. |
| | 5 | send | RF | TIS_AC_AIR_a (s:= add_H; tivid:=0; call:= BA1234; cat:= 2) | TAa | Send to the station under test an aircraft target (airborne) message for the same aircraft from a ground station H referring to TIV 0. |
| | 6 | record | AppOut | TIVa:= TIV SPECIFIED IN TARGET INFO ON BA1234 FROM STATION G (s:= add_A) | | Record information passed from the station under test to the application concerning info from ground station G. |
| | 7 | record | AppOut | TIVb:= TIV SPECIFIED IN TARGET INFO ON BA1234 FROM STATION H (s:= add_A) | | Record information passed from the station under test to the application concerning info from ground station H. |
| | 8 | verify | | TIVa ≠ TIVb | | Verify that the TIVs are identified by the station under test as different TIVs. |
| postamble | 9 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | TIS_TIV_Unique_B | | | | |
|------------------|------|---|--------|--|-----|---|
| Purpose: | | To demonstrate that the same TIV ID in target messages referring to different aircraft, and received from different ground stations, are identified as separate TIVs. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | TIS_MAN_a (s:= add_G; tivid:=0; nvert:= 0; lat: = CPR_LAT(N 50,5 NM); long:= CPR_LON(E1,3 NM)) | TMa | Send a management message to define TIV 0 from simulated ground station G with centre at 50,5 NM North, 1,3 NM East. |
| | 3 | send | RF | TIS_AC_AIR_a (s:= add_G; tivid:=0; call:= BA1234; cat:= 2) | TAa | Send to the station under test an aircraft target (airborne) message from ground station G referring to TIV 0. |
| | 4 | send | RF | TIS_MAN_a (s:= add_H; tivid:=0; nvert:= 0; lat: = CPR_LON(N 52.3 NM); long:= CPR_LON(0)) | TMa | Send a management message to define TIV 0 from simulated ground station H with different centre coordinates. |
| | 5 | send | RF | TIS_AC_AIR_a (s:= add_H; tivid:=0; call:= LH4567; cat:= 2) | TAa | Send to the station under test an aircraft target (airborne) message for a different aircraft from ground station H referring to TIV 0. |
| | 6 | record | AppOut | TIVa:= TIV SPECIFIED IN TARGET INFO ON BA1234 FROM STATION G | | Record information passed from the station under test to the application concerning info from ground station G. |
| | 7 | record | AppOut | TIVb:= TIV SPECIFIED IN TARGET INFO ON LH4567 FROM STATION H | | Record information passed from the station under test to the application concerning info from ground station H. |
| | 8 | verify | | TIVa ≠ TIVb | | Verify that the TIVs are identified by the station under test as different TIVs. |
| postamble | 9 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: | | TIS_TIV_Unique_C | | | | |
|-----------------|------|---|--------|---|-----|---|
| Purpose: | | To demonstrate that the same TIV ID in target messages referring to different aircraft or vehicles on the ground, and received from different ground stations, are identified as separate TIVs. | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | send | RF | TIS_MAN_a (s:= add_G; tivid:=0; nvert:= 0; lat: = CPR_LAT(N 50,5 NM); long:= CPR_LON(E 1,3 NM)) | TMa | Send a management message to define TIV 0 from simulated ground station G with centre at 50,5 NM North, 1,3 NM East. |
| | 3 | send | RF | TIS_AC_GRD_a (s:= add_G; tivid:=0; call:= BA1234; cat:= 2) | TGa | Send to the station under test an aircraft target (ground) message from ground station G referring to TIV 0. |
| | 4 | send | RF | TIS_MAN_a (s:= add_H; tivid:=0; nvert:= 0; lat: = CPR_LAT(N 52.3 NM); long:= CPR_LON(0)) | TMa | Send a management message to define TIV 0 from simulated ground station H with different centre coordinates. |
| | 5 | send | RF | TIS_AC_GRD_a (s:= add_H; tivid:=0; call:= LH4567; cat:= 2) | TGa | Send to the station under test an aircraft target (ground) message for a different aircraft from ground station H referring to TIV 0. |
| | 6 | record | AppOut | TIVa:= TIV SPECIFIED IN TARGET INFO ON BA1234 FROM STATION G | | Record information passed from the station under test to the application concerning info from ground station G. |
| | 7 | record | AppOut | TIVb:= TIV SPECIFIED IN TARGET INFO ON LH4567 FROM STATION H | | Record information passed from the station under test to the application concerning info from ground station H. |
| | 8 | verify | | TIVa ≠ TIVb | | Verify that the TIVs are identified by the station under test as different TIVs. |

| Test Case Name: TIS_TIV_Unique_C | | | | | | |
|--|------|--------|--------|--|-----|---|
| Purpose: To demonstrate that the same TIV ID in target messages referring to different aircraft or vehicles on the ground, and received from different ground stations, are identified as separate TIVs. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| | 9 | send | RF | TIS_MAN_a (s:= add_G; tivid:=0; nvert:= 0; lat: = CPR_LAT(N 50,5 NM); long:= CPR_LON(E 1,3 NM) | TMa | Send a management message to define TIV 0 from simulated ground station G with centre at 50,5 NM N, 1,3 NM East. |
| | 10 | send | RF | TIS_GRD_VEH_a (s:= add_G; tivid:=0; tid:= 1000) | TVa | Send to the station under test a ground vehicle message from ground station G referring to TIV 0. |
| | 11 | send | RF | TIS_MAN_a (s:= add_H; tivid:=0; nvert:= 0; lat: = CPR_LAT(N 52.3 NM); long:= CPR_LON(0) | TMa | Send a management message to define TIV 0 from simulated ground station H with different centre coordinates. |
| | 12 | send | RF | TIS_GRD_VEH_a (s:= add_H; tivid:=0; tid:= 1001) | TVa | Send to the station under test a ground vehicle message for a different vehicle from ground station H referring to TIV 0. |
| | 13 | record | AppOut | TIVa:= TIV SPECIFIED IN TARGET INFO ON VEHICLE 1000 FROM STATION G | | Record information passed from the station under test to the application concerning info from ground station G. |
| | 14 | record | AppOut | TIVb:= TIV SPECIFIED IN TARGET INFO ON VEHICLE 1001 FROM STATION H | | Record information passed from the station under test to the application concerning info from ground station H. |
| | 15 | verify | | TIVa ≠ TIVb | | Verify that the TIVs are identified by the station under test as different TIVs. |
| postamble | 16 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: TIS_Manage_Rec_A | | | | | | |
|--|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received management message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | TIS_Manage_a (TIS_Manage_RF_IN_PARAMETERS(x)) | TMa | Send a Manage message to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | Manage MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | TIS_Manage_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: TIS_Air_Air_Rec_A | | | | | | |
|--|------|--------|--------|--|-----|---|
| Purpose: To demonstrate that a station will correctly process a received aircraft airborne target message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | TIS_AIR_AIR_A (TIS_AIR_AIR_RF_IN_PARAMETERS(x)) | TAa | Send an airborne aircraft target message to the station under test from a simulated ground station G. |
| | 3 | await | AppOut | AIRBORNE TARGET MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 4 | verify | AppOut | TIS_AIR_AIR_A_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: TIS_Air_Ground_Rec_A | | | | | | |
|--|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received aircraft target ground message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 2 | send | RF | TIS_AIR_GRD_a (TIS_AIR_GROUND_RF_IN_PARAMETERS(x)) | TGa | Send an aircraft ground target message to the station under test from a simulated ground station G. |
| | 3 | await | AppOut | AIRCRAFT TARGET GROUND MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 4 | verify | AppOut | TIS_AIR_GROUND_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: TIS_Ground_Veh_Rec_A | | | | | | |
|---|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received ground vehicle target message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 2 | send | RF | TIS_AIR_GRD_a (TIS_GROUND_VEH_RF_IN_PARAMETERS(x)) | TVa | Send a ground vehicle message to the station under test from a simulated ground station G. |
| | 3 | await | AppOut | GROUND VEHICLE TARGET MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 4 | verify | AppOut | TIS_GROUND_VEH_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 5 | endrep | | next x | | Go to next parameter set. |
| postamble | 6 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

7.4.3.2.2.2

FIS-B test cases

| Test Case Name: FIS_Request_A | | | | | | |
|--|------|--------|-------|---|-----|--|
| Purpose: To demonstrate that a station will transmit a FIS-B report request message with the null reservation field with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 4 | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters |
| | 3 | send | AppIn | REQUEST TO TRANSMIT FIS-B REQUEST MESSAGE WITH NULL RESERVATION (FIS_REQUEST_APPIN_PARAMETERS(x)) | | Instruct the station under test to transmit a report request message with a null reservation, to request a particular FIS-B message. |
| | 4 | await | RF | BURST transmitted by the station under test | | Wait for a burst to be broadcast by the station under test. |
| | 5 | verify | RF | BURST (s:= add_A) has the format of "FIS_Request_a" | FQa | Verify that the transmitted burst is the same as "FIS_Request_a" set out in burst format clause. |
| | 6 | verify | RF | FIS_REQUEST_RF_OUT_PARAMETERS(x) | | Verify that the content of the information output by the station under test at the RF PCO is correct. |
| | 7 | endrep | | next x | | End loop. |
| postamble | 8 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: FIS_Request_B | | | | | | |
|---|------|--------|-------|--|-----|---|
| Purpose: To demonstrate that a station will transmit a FIS-B report request message with the unicast reservation field with the correct format. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep 4 | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters |
| | 3 | send | AppIn | REQUEST TO TRANSMIT FIS-B REQUEST MESSAGE WITH UNICAST RESERVATION (FIS_REQUEST_APPIN_PARAMETERS(x)) | | Instruct the station under test to transmit a report request message with a unicast reservation, to request a particular FIS-B message. |
| | 4 | await | RF | BURST transmitted by the station under test | | Wait for a burst to be transmitted by the station under test. |
| | 5 | verify | RF | BURST (s:= add_A) has the format of "FIS_Request_b" | FQb | Verify that the transmitted burst is the same as "FIS_Request_b" set out in burst format clause. |
| | 6 | verify | RF | FIS_REQUEST_RF_OUT_PARAMETERS(x) | | Verify that the content of the information output by the station under test at the RF PCO is correct. |
| | 7 | endrep | | next x | | End loop. |
| postamble | 8 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: FIS_METAR_Rec_A | | | | | | |
|---|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received METAR message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | FIS_METAR_a (FIS_METAR_RF_IN_PARAMETERS(x)) | FMa | Send a METAR message to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | METAR MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | FIS_METAR_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: FIS_SPECI_REC_A | | | | | | |
|---|------|--------|--------|--|-----|---|
| Purpose: To demonstrate that a station will correctly process a received SPECI message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | FIS_SPECI_a (FIS_SPECI_RF_IN_PARAMETERS(x)) | FSa | Send a SPECI message to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | SPECI MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | FIS_SPECI_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: FIS_ATIS_Rec_A | | | | | | |
|--|------|--------|--------|--|-----|---|
| Purpose: To demonstrate that a station will correctly process a received ATIS message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | FIS_ATIS_a (FIS_ATIS_RF_IN_PARAMETERS(x)) | FAa | Send a ATIS message to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | ATIS MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | FIS_ATIS_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: FIS_RCN_Rec_A | | | | | | |
|---|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received RCN message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | FIS_RVR_a (FIS_RVR_RF_IN_PARAMETERS(x)) | FRa | Send a RVR message to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | RVR MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | FIS_RVR_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: FIS_SIGMET_Rec_A | | | | | | |
|--|------|--------|--------|---|-----|---|
| Purpose: To demonstrate that a station will correctly process a received SIGMET message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | FIS_SIGMET_a (FIS_SIGMET_RF_IN_PARAMETERS(x)) | FSa | Send a SIGMET message to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | SIGMET MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | FIS_SIGMET_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: FIS_TSA_CTR_Rec_A | | | | | | |
|---|------|--------|--------|--|------|---|
| Purpose: To demonstrate that a station will correctly process a received TSA message for control zones. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | FIS_TSA_CTR_A (FIS_TSA_CTR_A RFINPARAMETERS(x)) | FTCa | Send a TSA message for control zones to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | TSA MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | FIS_TSA_CTR_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: FIS_TSA_DNG_Rec_A | | | | | | |
|--|------|--------|--------|--|------|--|
| Purpose: To demonstrate that a station will correctly process a received TSA message for restricted, danger or prohibited zones. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | FIS_TSA_DNG_A (FIS_TSA_DNG_A RFINPARAMETERS(x)) | FTDa | Send a TSA message for restricted, danger or prohibited zones to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | TSA MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | FIS_TSA_DNG_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

7.4.3.2.2.3

GNS-B test cases

| Test Case Name: GNS_Type1_Message_Rec_A | | | | | | |
|--|------|--------|--------|--|-----|---|
| Purpose: To demonstrate that a station will correctly process a received GNS-B Type 1 message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | GNS_TYPE1_a (GNS_TYPE1_RF_IN_PARAMETERS(x)) | G1a | Send a GNS-B message to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | GNS MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | GNS_TYPE1_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: GNS_Type2_Message_Rec_A | | | | | | |
|--|------|--------|--------|--|-----|---|
| Purpose: To demonstrate that a station will correctly process a received GNS-B Type 2 message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | GNS_TYPE2_a (GNS_TYPE2_RF_IN_PARAMETERS(x)) | G2a | Send a GNS-B message to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | GNS MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | GNS_TYPE2_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | Bring test equipment into idle state. |
| Comments: | | | | | | |

| Test Case Name: GNS_Type4_Message_Rec_A | | | | | | |
|--|------|--------|--------|--|-----|---|
| Purpose: To demonstrate that a station will correctly process a received GNS-B Type 4 message. | | | | | | |
| Context | Step | Action | PCO | Action Qualifier | Ref | Comment |
| preamble | 1 | do | | M_POWER_UP | | Prepare the transceiver for testing. |
| test body | 2 | rep x | | x:= {P, Q, R, S} | | Repeat test for 4 different sets of parameters. |
| | 3 | send | RF | GNS_TYPE4_a (GNS_TYPE4_RF_IN_PARAMETERS(x)) | G4a | Send a GNS-B message to the station under test from a simulated ground station G. |
| | 4 | await | AppOut | GNS MESSAGE INFORMATION | | Wait for the message information to be output by station under test at the AppOut PCO. |
| | 5 | verify | AppOut | GNS_TYPE4_APPOUT_PARAMETERS(x) | | Verify that the content of the information output by station under test at the AppOut PCO is correct. |
| | 6 | endrep | | next x | | Go to next parameter set. |
| postamble | 7 | | | | | 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:

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 implementor 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 implementor 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 airborne 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.

Table A.1: VDL Mode 4 requirements for ADS-B

| Requirement reference | Reference in [1] | Req | Title | Test Case |
|-----------------------|------------------|-----|--|--|
| 5.1.1 | 3.3 | | Information Field Formats | See note 1 |
| 5.1.1.1 | 3.3 | a | | See note 1a |
| 5.1.1.2 | 3.3.2 | a | | ADSB_Basic_A ADSB_Basic_Rec_A ADSB_High_Dynamic_A ADSB_High_Dynamic_Rec_A ADSB_Full_Position_A ADSB_Full_Position_Rec_A ADSB_Basic_Ground_Rec_A ADSB_UTC_Time_Rec_A ADSB_Two_Slot_TCP/SVQ_A ADSB_Two_Slot_TCP/SVQ_Rec_A ADSB_Single_Slot_TCP_A ADSB_Single_Slot_TCP_Rec_A ADSB_Single_Slot_SVQ_A ADSB_Single_Slot_SVQ_Rec_A ADSB_Aircraft_Data_A ADSB_Aircraft_Data_Rec_A ADSB_High_Resolution_A ADSB_High_Resolution_Rec_A CPR_Encode_Offset CPR_Decode_Offset |
| 5.1.1.3 | 3.3.3 | a | | See note 1a |
| 5.1.1.4 | 3.3.1 | a | | See note 1a |
| 5.1.1.5 | 3.3.3.2 | a | | See note 1a |
| 5.1.1.6 | 3.3.3.2 | b | | See note 1a |
| 5.1.1.7 | 3.3.4 | a | | See note 1a |
| 5.1.1.8 | 3.3.5 | a | | See note 1a |
| 5.1.1.9 | 3.3.6 | a | | See note 1a |
| 5.1.1.10 | 3.3.7 | a | | See note 1a |
| 5.1.1.11 | 3.3.8 | a | | See note 1a |
| 5.1.1.12 | 3.3.9 | a | | See note 1a |
| 5.1.2 | 3.4 | | ADS-B Request | See note 1 |
| 5.1.2.1 | 3.4.1 | | ADS-B Request Format | See note 1 |
| 5.1.2.1.1 | 3.4.1 | a | | ADSB_Request_A |
| 5.1.2.1.2 | 3.4.1 | b | | ADSB_Request_A ADSB_Request_B ADSB_Request_C ADSB_Request_D |
| 5.1.2.1.3 | 3.4.1 | c | | See note 1a |
| 5.1.2.1.4 | 3.4.1 | d | | See note 1a |
| 5.1.2.1.5 | 3.4.1 | e | | See note 1a |
| 5.1.2.2 | 3.4.2 | | Sleep Mode | See note 1 |
| 5.1.2.2.1 | 3.4.2 | a | | ADSB_Request_C ADSB_Request_D |
| 5.1.2.2.2 | 3.4.2 | b | | ADSB_Request_A ADSB_Request_B |
| 5.1.2.3 | 3.4.3 | | Automatic Selection of Variable Information Fields | See note 1 |
| 5.1.2.3.1 | 3.4.3 | a | | ADSB_Request_A ADSB_Request_C |
| 5.1.2.3.2 | 3.4.3 | b | | ADSB_Request_A ADSB_Request_C |
| 5.1.2.3.3 | 3.4.3 | c | | ADSB_Request_B ADSB_Request_D |
| 5.1.2.3.4 | 3.4.3 | d | | ADSB_Request_B ADSB_Request_D |
| 5.1.3 | 3.5 | | Default ADS-B Reporting | See note 1 |
| 5.1.3.1 | 3.5 | a | | ADSB_reporting |
| 5.1.3.1a | | | | See note 2 |
| 5.1.3.1b | | | | See note 2 |
| 5.1.3.2 | 3.5.1 | a | | See note 4 |

| Requirement reference | Reference in [1] | Req | Title | Test Case |
|-----------------------|------------------|-----|------------------------------------|--|
| 5.1.4 | 3.6 | | ADS-B Procedures | See note 1 |
| 5.1.4.1 | 3.6.1.1 | a | | ADSB_Request_B ADSB_Request_D |
| 5.1.4.2 | 3.6.1.1 | b | | ADSB_Request_A ADSB_Request_C |
| 5.1.4.3 | 3.6.1.1 | c | | ADSB_Request_A ADSB_Request_B |
| 5.1.4.4 | 3.6.1.2 | a | | ADSB_Request_C ADSB_Request_D |
| 5.1.4.4a | 3.6.1.2 | a | | See note 2 |
| 5.1.4.4b | 3.6.1.3 | a | | ADSB_Sleep_B |
| 5.1.4.4c | 3.6.1.3 | b | | ADSB_Sleep_B |
| 5.1.4.4d | 3.6.1.3 | c | | ADSB_Sleep_A |
| 5.1.4.5 | 3.6.2 | a | | ADSB_request_time |
| 5.1.4.6 | 3.6.3 | a | | ADSB_priority |
| 5.1.4.6a | 3.6.4.1 | a | | ADSB_First_TCP_A |
| 5.1.4.6b | 3.6.4.1 | b | | ADSB_First_TCP_A |
| 5.1.4.6c | 3.6.4.1 | c | | ADSB_First_TCP_A |
| 5.1.4.6d | 3.6.4.2 | a | | ADSB_Status_SVQ_A |
| 5.1.4.6e | 3.6.4.2 | b | | ADSB_Status_SVQ_A |
| 5.1.4.6f | 3.6.4.3 | a | | ADSB_TCP/SVQ_Flag_A |
| 5.1.4.7 | 3.6.4.4 | a | | ADSB_Status_SVQ_A |
| 5.1.4.8 | 3.6.4.5 | a | | ADSB_TCP/SVQ_Flag_A |
| 5.1.4.9 | 3.6.4.6 | a | | ADSB_TCP/SVQ_Flag_B |
| 5.1.5 | 3.7 | | CTRL Parameters | See note 1 |
| 5.1.5.1 | 3.7.1 | | Directory of Service (DOS) Message | See note 1 |
| 5.1.5.1.1 | 3.7.1.1 | a | | See note 3 |
| 5.1.5.1.2 | 3.7.1.1 | b | | See note 1a |
| 5.1.5.1.3 | 3.7.1.2 | a | | See note 1a |
| 5.1.5.2 | 3.7.2 | | Channel Management Parameter | See note 1 |
| 5.1.5.2.1 | 1.5.3.6.7 | a | | See note 1a |
| 5.1.5.2.2 | 1.5.3.6.7 | b | | See note 1a |
| 5.1.5.2.3 | 3.7.2 | a | | See note 1a |
| 5.1.5.2.4 | 3.7.2 | b | | See note 1a |
| 5.1.5.2.5 | 3.7.2.1.1 | a | | See note 1a |
| 5.1.5.2.6 | 3.7.2.1.2 | a | | See note 1a |
| 5.1.5.2.7 | 3.7.2.1.2 | b | | See note 1a |
| 5.1.5.2.8 | 3.7.2.2 | a | | See note 1a |
| 5.1.5.2.9 | 3.7.2.2.1 | a | | Channel_Dest_Count |
| 5.1.5.2.10 | 3.7.2.2.2 | a | | Channel_Dir_Rate_A Channel_Dir_Rate_B Channel_Directed |
| 5.1.5.2.10a | 3.7.2.2.2 | b | | Channel_Address |
| 5.1.5.2.11 | 3.7.2.2.3.1 | a | | Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.12 | 3.7.2.2.3.2 | a | | Channel_Dest_Count |
| 5.1.5.2.13 | 3.7.2.2.4.1 | a | | Channel_Directed |
| 5.1.5.2.14 | 3.7.2.2.4.1 | b | | Channel_Directed |
| 5.1.5.2.16 | 3.7.2.2.5 | a | | Channel_Multicast_A Channel_Multicast_B Channel_Multicast_C Channel_Multicast_D |
| 5.1.5.2.16a | 3.7.2.2.5 | b | | Channel_Multicast_A |
| 5.1.5.2.17 | 3.7.2.2.5 | c | | Channel_Multicast_B Channel_Multicast_C Channel_Multicast_D |
| 5.1.5.2.17a | 3.7.2.2.5 | d | | Channel_Multicast_B Channel_Multicast_C Channel_Multicast_D |
| 5.1.5.2.17b | 3.7.2.3.1.1 | a | | Channel_Dir_Rate_A Channel_Dir_Rate_B Channel_Directed |
| 5.1.5.2.17c | 3.7.2.3.1.1 | b | | Channel_Dir_Rate_A Channel_Directed |

| Requirement reference | Reference in [1] | Req | Title | Test Case |
|-----------------------|------------------|-----|-------|---|
| 5.1.5.2.17d | 3.7.2.3.1.1 | c | | Channel_Dir_Rate_B |
| 5.1.5.2.17e | 3.7.2.3.1.1 | d | | Channel_Invalid |
| 5.1.5.2.17f | 3.7.2.3.1.1 | e | | Channel_Invalid |
| 5.1.5.2.17g | 3.7.2.3.1.2 | a | | Channel_Dir_Rate_A Channel_Dir_Rate_B Channel_Directed |
| 5.1.5.2.17h | 3.7.2.3.1.2 | b | | See note 1a |
| 5.1.5.2.17i | 3.7.2.3.2 | a | | Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.17j | 3.7.2.3.2 | b | | Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.17k | 3.7.2.3.2 | c | | Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.17l | 3.7.2.3.3 | a | | Channel_Directed |
| 5.1.5.2.17m | 3.7.2.3.3 | b | | Channel_Directed |
| 5.1.5.2.17n | 3.7.2.3.3 | c | | Channel_Directed |
| 5.1.5.2.17o | 3.7.2.3.4 | a | | Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.17p | 3.7.2.3.4 | b | | Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.17q | 3.7.2.3.4 | c | | Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.17r | 3.7.2.3.4 | d | | Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.17s | 3.7.2.3.4 | e | | Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.17t | 3.7.2.3.4 | f | | Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.17u | 3.7.2.3.5.1 | a | | Channel_Dir_Rate_B |
| 5.1.5.2.17v | 3.7.2.3.5.1 | b | | Channel_Frequency |
| 5.1.5.2.17w | 3.7.2.3.5.1 | c | | Channel_Frequency |
| 5.1.5.2.17x | 3.7.2.3.5.1 | d | | Channel_Length |
| 5.1.5.2.17y | 3.7.2.3.5.2 | e | | Channel_Plea |
| 5.1.5.2.18 | 3.7.2.4.1 | a | | Channel_Cancel_DT |
| 5.1.5.2.19 | 3.7.2.4 | c | | Channel_Cancel_CSID_A Channel_Cancel_CSID_B |
| 5.1.5.2.19a | 3.7.2.4 | d | | Channel_Cancel_DT Channel_Cancel_CSID_A Channel_Cancel_CSID_B |
| 5.1.5.2.19b | 3.7.2.5 | a | | Channel_Cancel_Auto |
| 5.1.5.2.19c | 3.7.2.5.1.1 | a | | Channel_Cancel_CSID_C |
| 5.1.5.2.19d | 3.7.2.5.1.2 | a | | Channel_Cancel_CSID_D |
| 5.1.5.2.19e | 3.7.2.5.1.3 | a | | Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.19f | 3.7.2.5.1.4 | a | | Channel_Cancel_CSID_A Channel_Cancel_CSID_B |
| 5.1.5.2.19g | 3.7.2.5.1.5 | a | | Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B |
| 5.1.5.2.20 | 3.7.2.6.1 | a | | Channel_Directed Channel_Dir_Rate_A |
| 5.1.5.2.20a | 3.7.2.6.1 | b | | Channel_Unrecognize |
| 5.1.5.2.21 | 3.7.2.6.2 | a | | Channel_User_Trans Channel_Non_Sync_Trans |
| 5.1.5.2.22 | 3.7.2.6.2 | b | | Channel_User_Trans Channel_Non_Sync_Trans |
| 5.1.5.2.22a | 3.7.2.6.2.2 | a | | Channel_Speed |
| 5.1.5.2.22b | 3.7.2.6.2.3 | a | | Channel_NotSupported |

| Requirement reference | Reference in [1] | Req | Title | Test Case |
|-----------------------|------------------|-----|--|---|
| 5.1.5.2.22c | 3.7.2.7 | a | | Channel_Frequency |
| 5.1.5.2.22d | 3.7.2.7 | b | | See note 2 |
| 5.1.5.2.23 | 3.7.2.7 | c | | Channel_Frequency |
| 5.1.5.2.23a | 3.7.2.7.1 | a | | See note 4 |
| 5.1.5.2.24 | 3.7.2.8 | a | | Channel_Param_Q1 Channel_Param_Q4 Channel_Param_V12 Channel_Param_VS4 Channel_Param_VS2 |
| 5.1.5.2.24a | 3.7.2.8 | b | | See note 2 |
| 5.1.5.2.24b | 3.7.2.8 | c | | See note 2 |
| 5.1.5.2.24c | 3.7.2.8 | d | | See note 2 |
| 5.1.5.2.25 | 3.7.2.9.1 | a | | Channel_Dir_Rate_A Channel_Dir_Rate_B Channel_User_Trans |
| 5.1.5.2.26 | 3.7.2.9.2 | b | | Channel_Dir_Rate_A Channel_Dir_Rate_B Channel_User_Trans |
| 5.1.5.2.27 | 3.7.2.9.2 | c | | Channel_Script_Repeat |
| 5.1.5.2.28 | 3.7.2.9.2 | d | | See note 2 |
| 5.1.5.2.29 | 3.7.2.9.2 | e | | Channel_Discard |
| 5.1.6 | 4 | | Definitions for Compact Position Reporting | See note 1 |
| 5.1.6.1 | 4.6 | | Information Field Offset Encoding | See note 1 |
| 5.1.6.1.1 | 4.6.1 | a | | CPR_Encode_Offset |
| 5.1.6.1.2 | 4.6.1 | b | | CPR_Encode_Offset |
| 5.1.6.1.3 | 4.6.2 | a | | See note 1a |
| 5.1.6.1.4 | 4.6.3.1 | a | | CPR_Encode_Offset |
| 5.1.6.1.5 | 4.6.3.2 | a | | CPR_Encode_Offset |
| 5.1.6.1.6 | 4.6.3.3 | a | | CPR_Encode_Offset |
| 5.1.6.2 | 4.7 | | Information Field Offset Decoding | See note 1 |
| 5.1.6.2.1 | 4.7.1 | a | | CPR_Decode_Offset |
| 5.1.6.2.2 | 4.7.2 | a | | See note 1a |
| 5.1.6.2.3 | 4.7.3.1 | a | | CPR_Decode_Offset |
| 5.1.6.2.4 | 4.7.3.2 | a | | CPR_Decode_Offset |
| 5.1.6.3 | 4.8 | | Patch ID Encoding | See note 1 |
| 5.1.6.3.1 | 4.8.1 | a | | CPR_Encode_Offset |
| 5.1.6.3.2 | 4.8.2 | a | | See note 1a |
| 5.1.6.3.3 | 4.8.3 | a | | See note 1a |
| 5.1.6.4 | 4.9 | | Patch ID Decoding | See note 1 |
| 5.1.6.4.1 | 4.9.1 | a | | CPR_Decode_Offset |
| 5.1.6.4.2 | 4.9.2 | a | | See note 1a |
| 5.1.6.4.3 | 4.9.3.1 | a | | CPR_Decode_Offset |
| 5.1.6.4.4 | 4.9.3.2 | a | | CPR_Decode_Offset |
| 5.1.7 | | | Encoding of UDATA (udid) | See note 1 |
| 5.1.7.1 | | | | See note 1a |

Table A.2: VDL Mode 4 requirements for TIS-B

| Requirement reference | Title | Test Case |
|-----------------------|---|---|
| 5.2 | Requirements for TIS-B | See note 1 |
| 5.2.1 | Traffic information volume | See note 1 |
| 5.2.1.2 | | See note 1a |
| 5.2.1.3 | | See note 1a |
| 5.2.1.4 | | See note 1a |
| 5.2.1.5 | | See note 1a |
| 5.2.1.6 | | See note 1a |
| 5.2.1.7 | | See note 1a |
| | | |
| | | |
| 5.2.3 | Message format | See note 1 |
| 5.2.3.1 | | TIS_Manage_A TIS_Aircraft_Air_A TIS_Aircraft_Grd_A TIS_Grd_Veh_A |
| 5.2.3.2 | | TIS_Manage_A TIS_Aircraft_Air_A TIS_Aircraft_Grd_A TIS_Grd_Veh_A |
| 5.2.3.3 | | TIS_Manage_A TIS_Aircraft_Air_A TIS_Aircraft_Grd_A TIS_Grd_Veh_A |
| 5.2.3.4 | | See note 2 |
| 5.2.3.5 | | See note 2 |
| 5.2.3.6 | | TIS_Manage_A TIS_Aircraft_Air_A TIS_Aircraft_Grd_A TIS_Grd_Veh_A |
| 5.2.3.7 | | TIS_Manage_A TIS_Aircraft_Air_A TIS_Aircraft_Grd_A TIS_Grd_Veh_A |
| 5.2.3.8 | | TIS_Manage_A TIS_Aircraft_Air_A TIS_Aircraft_Grd_A TIS_Grd_Veh_A |
| 5.2.4 | Management message | See note 1 |
| 5.2.4.1 | | TIS_Manage_A |
| 5.2.4.2 | | TIS_Manage_A |
| 5.2.4.3 | | TIS_Manage_A |
| 5.2.4.4 | | TIS_Manage_A |
| 5.2.4.5 | | TIS_Manage_A |
| 5.2.5 | Aircraft target messages (airborne TIV) | See note 1 |
| 5.2.5.1 | | TIS_Aircraft_Air_A |
| 5.2.5.2 | | TIS_Aircraft_Air_A |
| 5.2.5.3 | | TIS_Aircraft_Air_A |
| 5.2.5.4 | | TIS_Aircraft_Air_A |
| 5.2.5.7 | | TIS_Aircraft_Air_A |
| 5.2.6 | Aircraft target messages (ground TIV) | See note 1 |
| 5.2.6.1 | | TIS_Aircraft_Grd_A |
| 5.2.6.2 | | TIS_Aircraft_Grd_A |
| 5.2.6.3 | | TIS_Aircraft_Grd_A |
| 5.2.6.4 | | TIS_Aircraft_Grd_A |
| 5.2.6.7 | | TIS_Aircraft_Grd_A |
| 5.2.7 | Ground vehicle target messages (ground TIV) | See note 1 |
| 5.2.7.1 | | TIS_Grd_Veh_A |
| 5.2.7.2 | | TIS_Grd_Veh_A |

| Requirement reference | Title | Test Case |
|-----------------------|---------------|-------------|
| 5.2.8 | Offset coding | See note 1 |
| 5.2.8.1 | | See note 1a |
| 5.2.8.2 | | See note 1a |
| 5.2.8.3 | | See note 1a |

Table A.3: VDL Mode 4 requirements for FIS-B

| Requirement reference | Title | Test Case |
|-----------------------|---|---|
| 5.3 | Requirements for FIS-B | |
| 5.3.1 | Message format | See note 1 |
| 5.3.1.1 | | FIS_METAR_A FIS_ATIS_A FIS_RCN_A FIS_SIGMET_A FIS_SPECI_A FIS_TSA_CTR_A FIS_TSA_DNG_A |
| 5.3.1.2 | | FIS_METAR_A FIS_ATIS_A FIS_RCN_A FIS_SIGMET_A FIS_SPECI_A FIS_TSA_CTR_A FIS_TSA_DNG_A |
| 5.3.1.3 | | FIS_METAR_A FIS_ATIS_A FIS_RCN_A FIS_SIGMET_A FIS_SPECI_A FIS_TSA_CTR_A FIS_TSA_DNG_A |
| 5.3.1.4 | | FIS_METAR_A FIS_ATIS_A FIS_RCN_A FIS_SIGMET_A FIS_SPECI_A FIS_TSA_CTR_A FIS_TSA_DNG_A |
| 5.3.1.5 | | FIS_METAR_A FIS_ATIS_A FIS_RCN_A FIS_SIGMET_A FIS_SPECI_A FIS_TSA_CTR_A FIS_TSA_DNG_A |
| 5.3.1.6 | | FIS_METAR_A FIS_ATIS_A FIS_RCN_A FIS_SIGMET_A FIS_SPECI_A FIS_TSA_CTR_A FIS_TSA_DNG_A |
| 5.3.2 | Meteorological Aerodrome Report (METAR) message | See note 1 |
| 5.3.2.1 | | FIS_METAR_A |
| 5.3.2.2 | | FIS_METAR_C |
| 5.3.2.3 | | FIS_METAR_A |
| 5.3.2.4 | | FIS_METAR_A |
| 5.3.2.5 | | FIS_METAR_A |
| 5.3.2.6 | | FIS_METAR_A |
| 5.3.2.7 | | FIS_METAR_A |
| 5.3.2.8 | | FIS_METAR_A |
| 5.3.2.9 | | FIS_METAR_A |

| Requirement reference | Title | Test Case |
|-----------------------|---|--------------------------------|
| 5.3.3 | Special Observations and Reports (SPECI) message | See note 1 |
| 5.3.3.1 | | FIS_SPECI_A |
| 5.3.4 | Automatic Terminal Information Service (ATIS) message | See note 1 |
| 5.3.4.1 | | FIS_ATIS_A |
| 5.3.4.2 | | FIS_ATIS_A |
| 5.3.4.3 | | FIS_ATIS_A |
| 5.3.4.4 | | FIS_ATIS_A |
| 5.3.4.5 | | FIS_ATIS_A |
| 5.3.4.6 | | FIS_ATIS_A |
| 5.3.4.7 | | FIS_ATIS_A |
| 5.3.5 | Runway Condition (RCN) message | See note 1 |
| 5.3.5.1 | | FIS_RCN_A |
| 5.3.5.2 | | FIS_RCN_A |
| 5.3.6 | SIGMET Reports | See note 1 |
| 5.3.6.1 | | FIS_SIGMET_A |
| 5.3.6.2 | | FIS_SIGMET_A |
| 5.3.6.3 | | FIS_SIGMET_A |
| 5.3.6.4 | | FIS_SIGMET_A |
| 5.3.7 | Temporary Segregated Areas (TSA) message | See note 1 |
| 5.3.7.1 | | FIS_TSA_CTR_A FIS_TSA_DNG_A |
| 5.3.7.2 | | FIS_TSA_CTR_A FIS_TSA_DNG_A |
| 5.3.7.3 | | FIS_TSA_CTR_A |
| 5.3.7.4 | | FIS_TSA_DNG_A |
| 5.3.7.5 | | FIS_TSA_CTR_A FIS_TSA_DNG_A |
| 5.3.8 | FIS report request message | See note 1 |
| 5.3.8.1 | | FIS_Request_Rec_A |
| 5.3.8.2 | | FIS_Request_Rec_A |
| 5.3.8.3 | | FIS_Request_Rec_A |
| 5.3.8.4 | | FIS_Request_Rec_A |
| 5.3.8.5 | | FIS_Request_Rec_A |
| 5.3.8.6 | | FIS_Request_Rec_A |
| 5.3.8.7 | | FIS_Request_Rec_A |
| 5.3.9a | Airborne system functions | See note 1 |
| | | |
| 5.3.9a.2 | Transmission of requests | See note 1 |
| 5.3.9a.2.1 | | See note 2 |
| 5.3.9a.3 | Differentiation between reports | See note 1 |
| 5.3.9a.3.1 | | See note 2 |

Table A.4: VDL Mode 4 requirements for GNS-B

| Requirement reference | Title | Test Case |
|-----------------------|--|---|
| 5.4 | Requirements for GNSS Augmentation Service Broadcast (GNS-B) message | See note 1 |
| 5.4.1 | Message format | See note 1 |
| 5.4.1.1 | | GNS_Type1_Message_A GNS_Type2_Message_A GNS_Type4_Message_A |
| 5.4.1.2 | | GNS_Type1_Message_A GNS_Type2_Message_A GNS_Type4_Message_A |
| 5.4.1.3 | | GNS_Type1_Message_A GNS_Type2_Message_A GNS_Type4_Message_A |
| 5.4.1.4 | | GNS_Type1_Message_A GNS_Type2_Message_A GNS_Type4_Message_A |
| 5.4.1.5 | | GNS_Type1_Message_A GNS_Type2_Message_A GNS_Type4_Message_A |
| 5.4.2 | Message Type 1 | See note 1 |
| 5.4.2.1 | | GNS_Type1_Message_A |
| 5.4.2.2 | | GNS_Type1_Message_A |
| 5.4.2.3 | | GNS_Type1_Message_A |
| 5.4.2.4 | | See note 2 |
| 5.4.2.5 | | See note 1a |
| 5.4.2.6 | | See note 2 |
| 5.4.3 | Message Type 2 | See note 1 |
| 5.4.3.1 | | GNS_Type2_Message_A |
| 5.4.3.2 | | GNS_Type2_Message_A |
| 5.4.4 | Message Type 4 | See note 1 |
| 5.4.4.1 | | GNS_Type4_Message_A |
| 5.4.4.2 | | GNS_Type4_Message_A |
| 5.4.4.3 | | GNS_Type4_Message_A |
| 5.4.4.3a | | See note 1a |
| 5.4.4.4 | | See note 2 |
| 5.4.4.5 | | See note 2 |
| 5.4.5 | CRC Calculation | See note 1 |
| 5.4.5.1 | GNS-B CRC | See note 1 |
| 5.4.5.1.1 | | See note 1a |
| 5.4.5.1.2 | | See note 1a |
| 5.4.5.1.3 | | See note 1a |
| 5.4.5.1.4 | | See note 1a |
| 5.4.5.1.5 | | See note 1a |
| 5.4.5.2 | FAS CRC | See note 1 |
| 5.4.5.2.1 | | See note 1a |
| 5.4.5.2.2 | | See note 1a |
| 5.4.5.2.3 | | See note 1a |
| 5.4.5.2.4 | | See note 1a |
| 5.4.5.2.5 | | See note 1a |
| 5.4.5.3 | Ephemeris CRC | See note 1 |
| 5.4.5.3.1 | | See note 1a |
| 5.4.5.3.2 | | See note 1a |
| 5.4.5.3.3 | | See note 1a |
| 5.4.5.3.4 | | See note 1a |
| 5.4.5.3.5 | | See note 1a |
| 5.4.5.3.6 | | See note 1a |
| 5.4.5.3.7 | | See note 1a |
| 5.4.5.3.8 | | See note 1a |
| 5.4.5.3.9 | | See note 1a |

Annex B (informative): Bibliography

- EUROCAE ED-108: "Interim MOPS for VDL Mode 4 Aircraft Transceiver for ADS-B", July 2001.
- North European ADS-B Network Update Programme: "FIS-B Service Description" version 1.5.1, 10th October 2003.
- North European ADS-B Network Update Programme: "TIS-B Service Description" version 1.33, 11th February 2003.
- Eurocontrol TIS-B Requirements, version 0.8a, 31st December 2002.
- North European ADS-B Network Update Programme: "GRAS Service Description" version 1.2, 19th November 2003.
- WMO Publication No306, Manual on Codes Vol 1.1, Part A for interpretation.
- EUROCAE ED-80 / RTCA DO-254 (April 2000): "Design Assurance Guidance for Airborne Electronic hardware".

NOTE: <http://www.eurocae.org>.

List of tables

| | |
|---|----|
| Table 5.1: ADS-B information fields | 16 |
| Table 5.2: Information field 0 hex - Basic..... | 16 |
| Table 5.3: Information field 1 hex - High dynamic | 17 |
| Table 5.4: Information field 2 hex - Full position | 17 |
| Table 5.5: Information field 3 hex - Basic ground | 17 |
| Table 5.6: Information field 4 hex - UTC time | 18 |
| Table 5.7: Information field 8 hex - Two slot TCP/SVQ..... | 18 |
| Table 5.8: Information field 9 hex - Single slot TCP | 19 |
| Table 5.9: Information field 5 hex - Single Slot SVQ..... | 19 |
| Table 5.10: Information field A1 hex - Aircraft data | 20 |
| Table 5.11: Information field AA0 hex - High resolution..... | 20 |
| Table 5.12: Information field encoding (variable data field)..... | 20 |
| Table 5.13: Baro/geo offset encoding (variable data field) | 22 |
| Table 5.14: Ground speed encoding (variable data field)..... | 23 |
| Table 5.15: Altitude rate magnitude encoding and decoding | 23 |
| Table 5.16: Aircraft category encoding..... | 24 |
| Table 5.17: Aircraft status (emergency/priority status) encoding | 24 |
| Table 5.18: Time to go subfield encoding | 24 |
| Table 5.19: ADS-B request bit encoding..... | 25 |
| Table 5.20: ADS-B request bit encoding for sleep mode parameters when sleep bit = 1 | 25 |
| Table 5.21: ADS-B request bit encoding for auto parameters when auto bit = 1..... | 26 |
| Table 5.22: ADS-B request field encoding..... | 26 |
| Table 5.22a: TCP update rate parameters..... | 28 |
| Table 5.22b: SVQ update rate parameters | 28 |
| Table 5.23: Encoding of application fields | 29 |
| Table 5.24: Encoding of service information field for sit = hex 00..... | 29 |
| Table 5.25: Component blocks of channel management parameter..... | 30 |
| Table 5.26: Header block bit encoding..... | 30 |
| Table 5.27: Header block field encoding..... | 31 |
| Table 5.28: Addressed destination block bit encoding..... | 32 |
| Table 5.29: Addressed destination block field encoding..... | 32 |
| Table 5.30: Regional multicast block bit encoding | 33 |
| Table 5.31: Regional multicast block field encoding | 33 |
| Table 5.32: Vertex distance encoding | 33 |
| Table 5.32a: Action on receipt of a channel management parameter..... | 34 |

| | |
|---|----|
| Table 5.33: Pre-defined transmission block definition encoding | 36 |
| Table 5.34: User-defined synchronization burst transmission block bit encoding | 36 |
| Table 5.35: User-defined synchronization burst transmission block field encoding | 36 |
| Table 5.36: Non-sync burst transmission block bit encoding | 36 |
| Table 5.37: Non-sync burst transmission block field encoding | 37 |
| Table 5.38: Frequency block bit encoding | 37 |
| Table 5.39: Frequency block field encoding | 37 |
| Table 5.40: Parameter block bit encoding | 38 |
| Table 5.41: Script block bit encoding | 38 |
| Table 5.42: Script block field encoding | 38 |
| Table 5.42a: Encoding of UDATA ID (udid) | 43 |
| Table 5.43: TIS-B message identifier encoding | 45 |
| Table 5.44: Management message bit encoding | 46 |
| Table 5.45: Management message field encoding | 47 |
| Table 5.46: Aircraft target message (airborne TIV) bit encoding (with callsign as an example flight ID) | 48 |
| Table 5.47: Aircraft target message (airborne TIV) field encoding | 49 |
| Table 5.48: Aircraft target message (ground TIV) bit encoding (with callsign as an example flight ID) | 50 |
| Table 5.49: Aircraft target message (ground TIV) field encoding | 50 |
| Table 5.50: Vehicle target message (ground TIV) bit encoding | 51 |
| Table 5.51: Ground vehicle target message (ground TIV) field encoding | 51 |
| Table 5.52: Values for scaling function corr(lat) used to encode target longitude | 52 |
| Table 5.53: Definition of offset encoding fields | 52 |
| Table 5.54: FIS-B message identifier encoding | 55 |
| Table 5.55: METAR message bit encoding | 56 |
| Table 5.56: METAR message field encoding | 57 |
| Table 5.57: Encoding for ICAO Weather codes | 61 |
| Table 5.57a: Free text 6-bit character encoding (converted from ASCII 8-bit) | 62 |
| Table 5.58: ATIS message bit encoding | 63 |
| Table 5.59: ATIS message field encoding | 64 |
| Table 5.60: RCN message bit encoding | 66 |
| Table 5.61: RCN message field encoding | 66 |
| Table 5.62: SIGMET message bit encoding | 69 |
| Table 5.63: SIGMET message field encoding | 70 |
| Table 5.64: TSA message fixed field bit encoding | 71 |
| Table 5.65: TSA message bit encoding for information field of type Control Zone | 71 |
| Table 5.66: TSA message bit encoding for Restricted, Danger, or Prohibited area | 71 |
| Table 5.67: TSA message field encoding | 72 |
| Table 5.68: FIS-B report request message bit encoding | 74 |

| | |
|--|-----|
| Table 5.69: FIS-B report request message field encoding..... | 74 |
| Table 5.70: GNS-B message identifier (gmi)..... | 75 |
| Table 5.71: GNS-B Message Type 1 bit encoding..... | 76 |
| Table 5.72: GNS-B Message Type 1 field encoding..... | 77 |
| Table 5.73: GNS-B message type 2 bit encoding..... | 80 |
| Table 5.74: GNS-B message type 2 field encoding..... | 81 |
| Table 5.75: GNS-B message type 4 bit encoding..... | 83 |
| Table 5.76: GNS-B message type 4 field encoding..... | 84 |
| Table 5.77: GPS Satellite ephemeris mask..... | 90 |
| Table 7.1: Protocol test-suite overview | 91 |
| Table 7.2: Subfield mnemonics | 94 |
| Table 7.3: Key to CPR encoding table in following clause..... | 99 |
| Table 7.4: Table of test values for CPR position report encoding CPR_ENC_TABLE (row, column) (CE(r, c)) | 100 |
| Table 7.5: CPR test value tolerances | 103 |
| Table 7.6: Key to CPR decoding table in following clause..... | 104 |
| Table 7.7: Table of test values for CPR position report decoding CPR_DEC_TABLE (row, column) (CD(r, c))..... | 105 |
| Table 7.8: Values for use in the test "ADSB_BASIC_A" | 108 |
| Table 7.9: Values for use in the test "ADSB_Basic_Rec_A"..... | 109 |
| Table 7.10: Values for use in the test "ADSB_High_Dynamic_A" | 110 |
| Table 7.11: Values for use in the test "ADSB_High_Dynamic_Rec_A" | 111 |
| Table 7.12: Values for use in the test "ADSB_Full_Position_A" | 112 |
| Table 7.13: Values for use in the test "ADSB_Full_Position_Rec_A" | 113 |
| Table 7.14: Values for use in the test "ADSB_Basic_Ground_Rec_A" | 114 |
| Table 7.15: Values for use in the test "ADSB_UTC_Time_Rec_A" | 115 |
| Table 7.16: Values for use in the test "ADSB_Two_Slot_TCP/SVQ_A"..... | 115 |
| Table 7.17: Values for use in the test "ADSB_Two_Slot_TCP/SVQ_Rec_A" | 118 |
| Table 7.18: Values for use in the test "ADSB_Single_Slot_TCP_A"..... | 120 |
| Table 7.19: Values for use in the test "ADSB_Single_Slot_TCP_Rec_A" | 121 |
| Table 7.20: Values for use in the test "ADSB_Single_Slot_SVQ_A" | 122 |
| Table 7.21: Values for use in the test "ADSB_Single_Slot_SVQ_Rec_A" | 123 |
| Table 7.22: Values for use in the test "ADSB_Single_Slot_SVQ_Rec_A" | 124 |
| Table 7.23: Values for use in the test "ADSB_Aircraft_Data_Rec_A" | 124 |
| Table 7.24: Values for use in the test "ADSB_High_Resolution_A" | 125 |
| Table 7.25: Values for use in the test "ADSB_High_Resolution_Rec_A" | 126 |
| Table 7.26: Values for use in the test "TIS_Manage_Rec_A" | 127 |
| Table 7.27: Values for use in the test "TIS_Air_Air_Rec_A"..... | 128 |
| Table 7.28: Values for use in the test "TIS_Air_Ground_Rec_A" | 129 |

| | |
|--|-----|
| Table 7.29: Values for use in the test "TIS_Ground_Veh_Rec_A" | 130 |
| Table 7.30: Values for use in the test "FIS_Metar_Rec_A" | 131 |
| Table 7.31: Values for use in the test "FIS_ATIS_Rec_A" | 133 |
| Table 7.32: Values for use in the test "FIS_RCN_Rec_A" | 135 |
| Table 7.33: Values for use in the test "FIS_SIGMET_REC_A" | 136 |
| Table 7.34: Values for use in the test "FIS_TSA_CTR_REC_A"..... | 137 |
| Table 7.35: Values for use in the test "FIS_TSA_DNG_REC_A"..... | 138 |
| Table 7.36: Values for use in the test "FIS_REQUEST_A"..... | 138 |
| Table 7.37: Values for use in the test "FIS_REQUEST_B" | 139 |
| Table 7.38: Values for use in the test "GNS_Type1_Message_Rec_A" | 139 |
| Table 7.39: Values for use in the test "GNS_Type2_Message_Rec_A" | 141 |
| Table 7.40: Values for use in the test "GNS_Type4_Message_Rec_A" | 142 |
| Table 7.41: SYNC_BASIC_a (SBa): Basic variable part - Occupies one slot..... | 144 |
| Table 7.42: SYNC_HIGH_DYNAMIC_a (SHa): High dynamic variable part - Occupies one slot..... | 144 |
| Table 7.43: SYNC_FULL_POSITION_a (SPa): Full Position variable part - Occupies one slot | 145 |
| Table 7.44: SYNC_BASIC_GROUND_a (BGa): Basic Ground variable part - Occupies one slot..... | 146 |
| Table 7.45: SYNC_UTC_TIME_a (UTCa): UTC Time variable part - Occupies one slot | 146 |
| Table 7.46: SYNC_TWO_SLOT_TCP/SVQ_a (STa): Two-slot TCP/SVQ variable part - Occupies two slots..... | 147 |
| Table 7.47: SYNC_SINGLE_SLOT_TCP_a (SSa): Single Slot TCP variable part - Occupies one slot..... | 148 |
| Table 7.48: SYNC_SINGLE_SLOT_SVQ_a (SVa): Single Slot SVQ variable part - Occupies one slot..... | 149 |
| Table 7.49: SYNC_AIRCRAFT_DATA_a (SDa): Aircraft data variable part - Occupies one slot | 149 |
| Table 7.50: SYNC_HIGH_RESOLUTION_a (SRa): High resolution variable part - Occupies one slot..... | 150 |
| Table 7.51: DIR_REQ_a (Da): Contains general request..... | 151 |
| Table 7.52: DIR_SYNC_BURST_a (DSa): Directed sync burst - Occupies one slot. Basic variable part.... | 151 |
| Table 7.53: PLEA_a (Pa): Information field contains destination address - Fits within delayed burst..... | 152 |
| Table 7.54: PLEA_RESP_a (PRa): Directed request with pr_flag = 1, nr ≠ "special" | 152 |
| Table 7.55: ADSB_REQUEST_a (ARa): Includes unicast reservation, sleep = 0, auto = 0..... | 153 |
| Table 7.56: ADSB_REQUEST_b (ARb): Includes unicast reservation, sleep = 0, auto = 1 | 153 |
| Table 7.57: ADSB_REQUEST_c (ARc): Includes directed request reservation, sleep = 1, auto = 0..... | 154 |
| Table 7.58: ADSB_REQUEST_d (ARd) Includes directed request reservation, sleep = 1, auto = 1 | 154 |
| Table 7.59: ADSB_REQUEST_e (ARe) Includes directed request reservation, sleep = 0, auto = 1..... | 155 |
| Table 7.60: DIR_SYNC_BURST_b (DSb): Directed sync burst - Occupies one slot - Full position variable part..... | 155 |
| Table 7.61: DIR_SYNC_BURST_c (DSc): Directed sync burst - Occupies one slot - Two-slot TCP/SVQ variable part | 156 |
| Table 7.62: CTRL_CHAN_a (Ca): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, no transmission block, 1 frequency block, no parameter block and 1 script block) - Fits within one slot | 157 |

| | |
|---|-----|
| Table 7.63: CTRL_CHAN_b (Cb): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 destination block, no transmission block, 1 frequency block, no parameter block and 1 script block) | 157 |
| Table 7.64: CTRL_CHAN_c (Cc): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 destination block, no transmission block, 1 frequency block, no parameter block and 2 script blocks)..... | 158 |
| Table 7.65: CTRL_CHAN_d (Cd): Channel management parameter in UCTRL burst with response reservation (1 header block, 2 destination blocks, no transmission block, 1 frequency block, no parameter block and 1 script block) | 159 |
| Table 7.66: CTRL_CHAN_e (Ce): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 multicast destination block, no transmission block, 1 frequency block, no parameter block and 1 script block) | 159 |
| Table 7.67: CTRL_CHAN_f (Cf): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 destination block, no transmission block, 2 frequency blocks, no parameter block and 2 script blocks)..... | 160 |
| Table 7.68: CTRL_CHAN_g (Cg): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, 1 transmission block, 1 frequency block, no parameter block and 2 script blocks) - Fits within one slot..... | 161 |
| Table 7.69: CTRL_CHAN_h (Ch): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, 2 transmission blocks, 1 frequency block, no parameter block and 2 script blocks) - Fits within one slot..... | 162 |
| Table 7.70: CTRL_CHAN_i (Ci): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, 2 transmission blocks (1 non-sync burst), 1 frequency block, no parameter block and 2 script blocks) | 162 |
| Table 7.71: CTRL_CHAN_j (Cj): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 destination block, 1 transmission block, 1 frequency block, no parameter block and 1 script block) | 163 |
| Table 7.72: CTRL_CHAN_k (Ck): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, 1 transmission block, 1 frequency block, no parameter block and 1 script blocks (with 1 repeat)) - Fits within one slot | 164 |
| Table 7.73: CTRL_CHAN_l (Cl): Channel management parameter in UCTRL burst with response reservation (1 header block, no destination block, 1 transmission block, 1 frequency block, no parameter block and 1 script blocks (with 2 invalid repeats)). Fits within one slot..... | 164 |
| Table 7.74: CTRL_CHAN_m (Cm): Channel management parameter in UCTRL burst with response reservation (1 header block, 1 destination block, no transmission block, 1 frequency block, 1 parameter block and 1 script block) | 165 |
| Table 7.75: SYNC_BURST_g (Sg): Occupies one slot. Autonomous burst - Basic variable part - Lat6 and lon6 specified. "x" = do not care | 166 |
| Table 7.76: SYNC_BURST_h (Sh): Occupies one slot - Autonomous burst, basic ground variable part Lat4, lon4, and pid specified. "x" = do not care | 166 |
| Table 7.77: SYNC_BURST_i (Si): Occupies one slot - Autonomous burst, high resolution variable part - Lat8 and lon8 specified. "x" = do not care..... | 167 |
| Table 7.78: TIS_MAN_a (TMa): Management message with response reservation field..... | 168 |
| Table 7.79: TIS_AC_AIR_a (TAa): Aircraft target message (airborne TIV) with response reservation field..... | 169 |
| Table 7.80: TIS_AC_GRD_a (TGa): Aircraft target message (ground TIV) with response reservation field..... | 170 |

| | |
|---|-----|
| Table 7.81: TIS_GRD_VEH_a (TVa): Ground vehicle target message with response reservation field..... | 171 |
| Table 7.82: FIS_METAR_a (FMa): METAR message with response reservation field..... | 171 |
| Table 7.83: FIS_SPECI_a (FSa): SPECI message with response reservation field | 172 |
| Table 7.84: FIS_ATIS_a (FAa): ATIS message with response reservation field..... | 173 |
| Table 7.85: FIS_RCN_a (FRa): RCN message with response reservation field | 174 |
| Table 7.86: FIS_SIGMET_a (FSa): SIGMET message with response reservation field | 175 |
| Table 7.87: FIS_TSA_CTR_a (FTCa): TSA CTR Zone message with response reservation field | 176 |
| Table 7.88: FIS_TSA_DNG_a(FTDa): TSA DNG Zone message with response reservation field | 177 |
| Table 7.89: FIS_REQUEST_a (FQa): Request message with null reservation field | 177 |
| Table 7.90: FIS_REQUEST_b (FQb): Request message with unicast reservation field..... | 178 |
| Table 7.91: GNS_TYPE1_a (G1a): GNS-B Type 1 message with response reservation | 178 |
| Table 7.92: GNS_TYPE2_a (G2a): GNS-B Type 2 message with response reservation | 179 |
| Table 7.93: GNS_TYPE4_a (G4a): GNS-B Type 4 message with response reservation | 180 |
| Table A.1: VDL Mode 4 requirements for ADS-B | 260 |
| Table A.2: VDL Mode 4 requirements for TIS-B | 264 |
| Table A.3: VDL Mode 4 requirements for FIS-B | 265 |
| Table A.4: VDL Mode 4 requirements for GNS-B | 267 |

History

| Document history | | | |
|-------------------------|----------------|----------------|---------------------------------------|
| V1.1.1 | September 2004 | Public Enquiry | PE 20050107: 2004-09-08 to 2005-01-07 |
| V1.1.1 | May 2005 | Vote | V 20050722: 2005-05-23 to 2005-07-22 |
| V1.1.1 | July 2005 | Publication | |
| | | | |
| | | | |