ETSI EN 302 842-3 V1.4.1 (2015-04)

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
REN/AERO-00019-3

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

The present document can be downloaded from:
http://www.etsi.org/standards-search

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

Users of the present document should be aware that the document may be subject to revision or change of status.
Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

Contents

Intellectual Property Rights 5
Foreword 5
Modal verbs terminology 5
Introduction 6
1 Scope 7
2 References 8
2.1 Normative references 8
2.2 Informative 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 12
4.1 General 12
4.2 Automatic Dependent Surveillance-Broadcast (ADS-B) 12
4.3 Traffic Information Service-Broadcast (TIS-B) 13
4.4 Flight Information Service-Broadcast (FIS-B) 14
4.5 GNSS Augmentation Service-Broadcast (GNS-B) 15
5 Minimum performance specification under standard test conditions 16
5.1 Requirements for ADS-B 16
5.1.1 Information Field Formats 16
5.1.2 ADS-B Request 26
5.1.2.1 ADS-B request format 26
5.1.2.2 Sleep Mode 26
5.1.2.3 Automatic Selection of Variable Information Fields 26
5.1.3 Default ADS-B Reporting 29
5.1.4 ADS-B Procedures 29
5.1.5 CTRL Parameters 31
5.1.5.1 Directory of Service (DOS) message 31
5.1.5.2 Channel Management Parameter 32
5.1.6 Definitions for Compact Position Reporting 42
5.1.6.1 Information Field Offset Encoding 42
5.1.6.2 Information Field Offset Decoding 44
5.1.6.3 Patch ID Encoding 45
5.1.6.4 Patch Id Decoding 45
5.1.7 Encoding of UDATA (udid) 46
5.2 Requirements for TIS-B 46
5.2.1 Traffic Information Volume (TIV) 46
5.2.2 Void 46
5.2.3 Message format 46
5.2.4 Management message 47
5.2.5 Aircraft target messages (airborne TIV) 49
5.2.6 Aircraft target messages (ground TIV) 51
5.2.7 Ground vehicle target messages (ground TIV) 52
5.2.8 TIS-B offset encoding 53
5.3 Requirements for FIS-B 56
5.3.1 Message format for received FIS-B messages 56
5.3.2 Meteorological Aerodrome Report (METAR) message 57
5.3.3 Special Observations and Reports (SPECI) message 64
5.3.4 Automatic Terminal Information Service (ATIS) message 65
5.3.5 Runway Condition (RCN) message 67

5.3.6 SIGnificant METeorological Information (SIGMET) message ... 71
5.3.7 Temporary Segregated Areas (TSA) message ... 72
5.3.8 FIS-B report request message ... 75
5.3.9 Void ... 76
5.3.9a Airborne system functions ... 76
5.3.9a.1 Transmission of requests ... 76
5.3.9a.2 Differentiation between reports ... 76
5.4 Requirements for GNSS Augmentation Service Broadcast (GNS-B) message ... 77
5.4.1 Message format ... 77
5.4.2 Message Type 1 ... 78
5.4.3 Message Type 2 ... 81
5.4.4 Message Type 4 ... 84
5.4.5 CRC Calculation ... 90
5.4.5.1 GNS-B CRC ... 90
5.4.5.2 FAS CRC ... 90
5.4.5.3 Ephemeris CRC ... 91

6 General design requirements ... 91

7 Protocol test procedures ... 91
7.1 General ... 91
7.2 Required test rig ... 92
7.3 Protocol test-suite description methodology ... 92
7.4 Detailed protocol test procedures ... 92
7.4.1 Test-suite overview ... 92
7.4.2 Declarations ... 95
7.4.3 Constraints ... 96
7.4.3.1 Abbreviations ... 96
7.4.3.1.1 Subfield mnemonics ... 96
7.4.3.1.2 Station addresses and positions ... 99
7.4.3.1.3 Tables of values for use in CPR test cases ... 100
7.4.3.1.4 Tables of values for use in content checking test cases ... 112
7.4.3.1.5 VDL4 Burst formats ... 146
7.4.3.2 Test cases ... 180
7.4.3.2.1 Test case macros ... 181
7.4.3.2.2 Test case descriptions ... 182

Annex A (informative): Cross reference matrix ... 261

Annex B (informative): Bibliography ... 271

History ... 272

Intellectual Property Rights

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

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

Foreword

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

The present document is part 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, ETSI EN 301 842 [i.7] parts 1 to 5, covering the VHF air-ground Data Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for ground-based equipment.

NOTE: Minimum Operational Performance Specifications (MOPS) are also being developed for VDL Mode 4. EUROCAE have previously published Interim MOPS for VDL Mode 4 [i.6] which are a sub set of ETSI EN 302 842-1 [1], ETSI EN 302 842-2 [2], ETSI EN 302 842-3 (the present document) and ETSI EN 302 842-4 [3]. ETSI EN 302 842-1 [1], ETSI EN 302 842-2 [2], ETSI EN 302 842-3 (the present document) and ETSI EN 302 842-4 [3] comply with the requirements of CEC Mandate M/318 [i.8].

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

Modal verbs terminology

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

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

Introduction

The present document states the technical specifications for Very High Frequency (VHF) Digital Link (VDL) Mode 4 aeronautical mobile (airborne) radio transmitters, transceivers and receivers for air-ground and air-air communications operating in the VHF band, using Gaussian Filtered Frequency Shift Keying (GFSK) modulation with 25 kHz channel spacing and capable of tuning to any of the 25 kHz channels from 112,000 MHz to 136,975 MHz as defined in ICAO VDL SARPs [i.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 complies with ETSI EN 302 842-1 [1] and ETSI EN 302 842-2 [2].

NOTE: The present document has been produced with a view to maintaining consistency of numbering with the equivalent standard for ground equipment (ETSI EN 301 842 [i.7] parts 1 to 4). Where requirements are the same, they have been given the same number. Some new airborne requirements have been inserted between requirements that were sequential in ETSI EN 301 842 [i.7] parts 1 to 4. This has led to a non-standard form of numbering for new requirements in some places.

1 Scope

The present document applies to the following radio equipment types:

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

The present document provides part 3 of the technical specifications.

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

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

The present document applies to "aeronautical mobile (airborne and 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 ETSI EN 301 842 [i.7].

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

ETSI EN 302 842-1 [1] deals with tests of the physical layer. ETSI EN 302 842-2 [2] 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:

- clause 2 provides references to relevant documents;
- clause 3 provides general definitions, abbreviations and symbols used;
- clause 4 describes the VDL Mode 4 mobile station 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 [i.1];
- annex B provides a Bibliography.

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

2.1 Normative references

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

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

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

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

[1] ETSI EN 302 842-1 (V1.3.1): "VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment; Part 1: Physical layer".

[2] ETSI EN 302 842-2 (V1.4.1): "VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment; Part 2: General description and data link layer".

[3] ETSI EN 302 842-4 (V1.3.1): "VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment; Part 4: Point-to-point functions".

[4] WMO Publication No. 306: "Manual on Codes Vol 1.1, Part A".

2.2 Informative references

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

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

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

[i.1] ICAO Doc 9816 (First Edition 2004): "Manual on VHF Digital Link (VDL) Mode 4 - Part 2: Detailed Technical Specifications".

[i.2] ICAO Annex 10 to the Convention on International Civil Aviation: "Aeronautical Telecommunications, Volume III: Communication Systems, Part I: Digital Data Communication Systems, chapter 6", including Amendment 88 (applicable 14/11/13).

[i.3] RTCA DO-242A: "Minimum Aviation System Performance Standards for Automatic Dependent Surveillance Broadcast (ADS-B)".

[i.4] 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.

[i.5] EUROCAE ED-114A: "Minimum Operational Performance Specification for Global Navigation Satellite Ground Based Augmentation System Ground Equipment to Support Category I Operations".

NOTE: http://www.eurocae.net.

[i.6] EUROCAE ED-108A: "MOPS for VDL Mode 4 Aircraft Transceiver for ADS-B", July 2001.

[i.7] ETSI EN 301 842 (all parts): "VHF air-ground Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for ground-based equipment".

[i.8] CEC Mandate M/318: "Mandate to CEN/CENELEC/ETSI for standardisation in the field of air traffic management systems and Galileo local components".

3 Definitions and abbreviations

3.1 Definitions

3.1.1 Basic reference model definitions

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

3.1.2 Service conventions definitions

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

3.1.3 General definitions

For the purposes of the present document, the terms and definitions given in ETSI EN 302 842-1 [1], clause 3.1.3, ETSI EN 302 842-2 [2], 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 are indicated by subscript numbers. Bit 1 indicates the least significant bit; and

b) bits are transmitted octet by octet, starting with the first octet in each table, and within each octet the rightmost bit (as shown in the tables) is transmitted first.

3.2 Abbreviations

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

ACAS	Airborne Collision Avoidance System
ADS	Automatic Dependent Surveillance
ADS-B	Automatic Dependent Surveillance-Broadcast
ATC	Air Traffic Control
ATIS	Automatic Terminal Information Service
ATM	Air Traffic Management
BND	Big Negative Dither
CCI	Co-Channel Interference
CDTI	Cockpit Display of Traffic Information
CPR	Compact Position Reporting
CRC	Cyclic Redundancy Code
CSA	Call Segment Association
CTRL	ConTRoL
dB	deciBel
DFPAP	Delta Flight Path Alignment Point
DLPDU	Data Link Protocol Data Unit
DLS	Data Link Service
DOS	Directory Of Services
DT	Directed Timeout
EGNOS	European Geostationary Navigation Overlay Service
EPU	Estimated Position Uncertainty
EUROCAE	EURopean Organization for Civil Aviation Equipment
FAS	Final Approach Segment
FIR	Flight Information Region
FIS	Flight Information Service

FIS-B	Flight Information Service-Broadcast
FPAP	Flight Path Alignment Point
fpm	feet per minute
fps	feet per second
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	International Alphabet
IA-5	International Alphabet-5
ICAO	International Civil Aviation Organization
ID	IDentity
ILS	Instrument Landing System
INFO	INFOrmation (DLPDU)
IRVR	Instrumented RVR
LSB	Least Significant Bit
LTP	Landing Threshold Point
MASPS	Minimum Aviation System Performance Standards
MEDUP	MEDiterranean Update Programme
METAR	METeorological Aerodrome Report
MLS	Microwave Landing System
MOPS	Minimum Operational Performance Specification
ms	milliseconds
MSAS	Multi-functional Satellite Augmentation System
MSB	Most Significant Bit
NAC	Network Access Credentials
NDB	Non-Directional Beacon
NEAN	North European ADS-B Network
NES	Network End System
NIC	Navigation Integrity Category
NM	Nautical Mile
NTM	NT1 Test Mode/Network Traffic Management
NUP	NEAN Update Programme
PCO	Point of Control and Observation
PRN	Pseudo Random Noise
RCM	Remote Command Message
RCN	Runway Condition
RF	Radio Frequency
RR	Receiver Position Vector
RVR	Routing Verification Result
SARPs	Standards And Recommended Practices
SBAS	Space-Based Augmentation System
SIGMET	SIGnificant METeorological information
SIL	Surveillance Integrity Level
SPECI	SPECIal observations and reports
SPI	Special Position Indicator
SST	Sea Surface Temperature
SVQ	State Vector Quality
TBD	To Be Defined
TCH	Threshold Crossing Height
TCP	Trajectory Change Point
TIS	Traffic Information Service
TIS-B	Traffic Information Service-Broadcast
TIV	Traffic Information Volume

TSA	Temporary Segregated Areas
TV	Transmission Volume
TWR	Tower
UCTRL	Unacknowledged ConTRoL data broadcast (DLPDU)
UTC	Universal Time Coordinated
VDF	VHF Direction Finding
VDL	VHF Digital Link
VEPU	Vertical Estimated Position Uncertainty
VHF	Very High Frequency
VOR	VHF Omnidirectional Radio
VSS	VDL Mode 4 Specific Services
WAAS	Wide Area Augmentation System
WGS	World Geodetic System

NOTE: As in WGS-84: World Geodetic System 1984.

WMO	World Meteorological Organization
XOR	Exclusive OR

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 ETSI EN 302 842-2 [2]. 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 ETSI EN 302 842-4 [3].

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 Dependent 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 of the 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, vehicles and ground subsystems. The capabilities of participant subsystems will vary based upon class of equipage. 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 the 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.

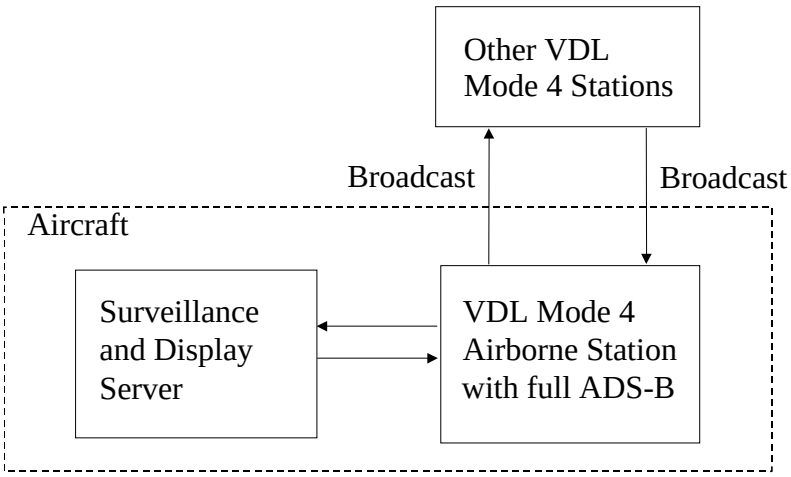

Figure 4.1: Airborne station supporting ADS-B functions

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 [i.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.

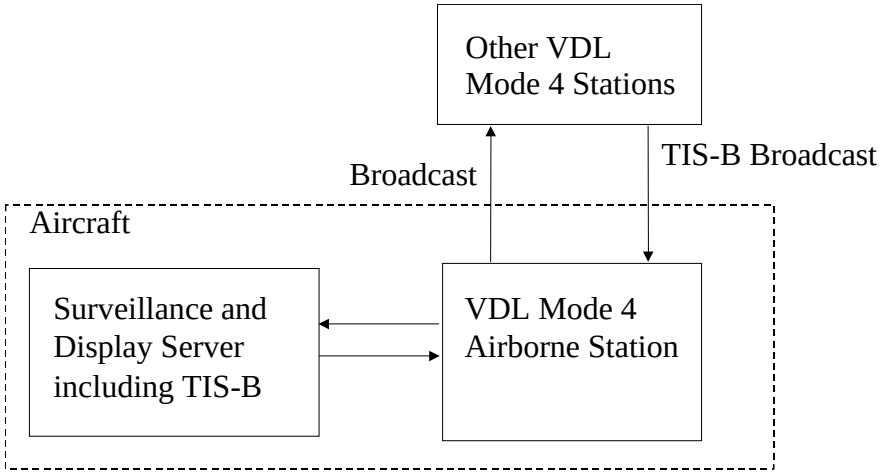

Figure 4.2: Airborne station supporting TIS-B functions

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): report indicating meteorological conditions at aerodromes, officially recoded 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): 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): 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): 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): 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.

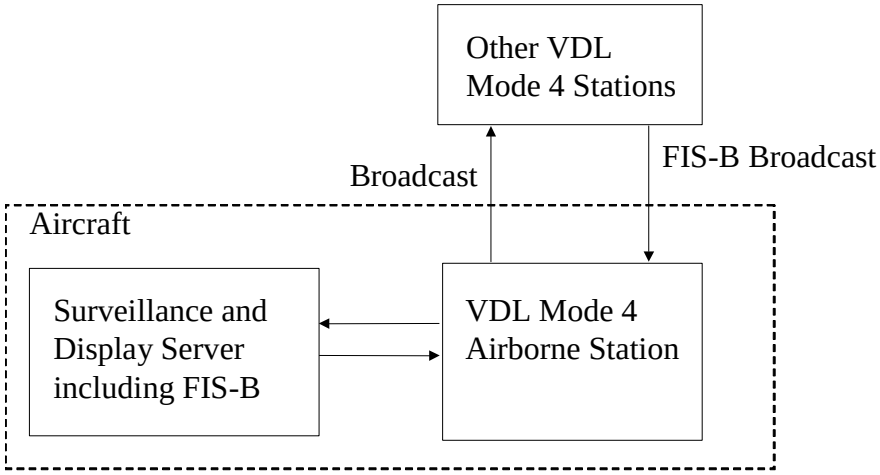

Figure 4.3: Ground station supporting FIS-B functions

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 Nean 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 [i.4], GBAS MOPS [i.5], 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.

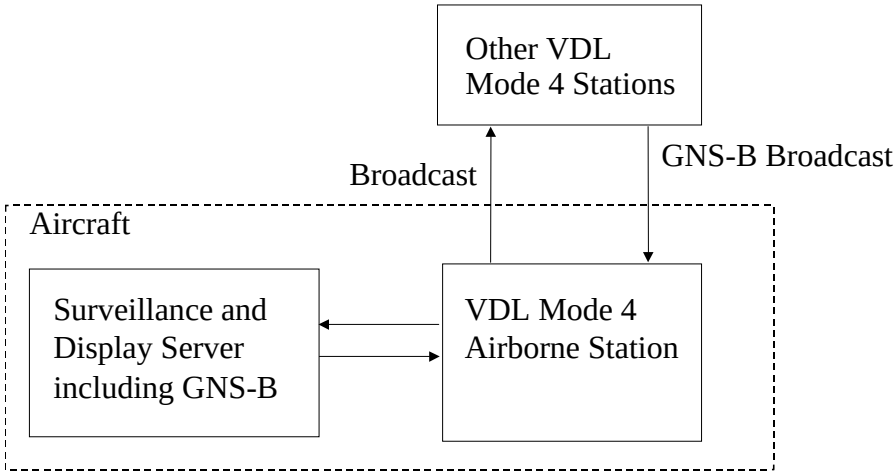

Figure 4.4: Airborne supporting GNS-B functions

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
6 hex	Not present	Not present	Aircraft ID data
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 tables 5.2 to 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	$lat6_6$	$lat6_5$	$lat6_4$	$lat6_3$	$lat6_2$	$lat6_1$
6-bit longitude offset (lon6) baro rate/geo rate (br/gr), special position indicator (spi)	13	spi	br/gr	$lon6_6$	$lon6_5$	$lon6_4$	$lon6_3$	$lon6_2$	$lon6_1$
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 (SILI)	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 (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: "res" denotes currently unused. "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

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
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}
Resolution Advisory active flag (ra) ACAS operational flag (acas), NIC_{baro}(nicb), CDTI display capability (cdti), NAC_p (nacp)	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) altitude rate (altr)	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; "res" denotes currently unused.
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 1: "x" denotes part of fixed data field.
NOTE 2: 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.

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
Reserved for future definition	18	res	res	res	res	res	res		

NOTE 1: "x" denotes part of fixed data field.
NOTE 2: "res" denotes currently unused.
NOTE 3: 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 5.10a: Information field 6 Hex - Aircraft ID data

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
Information field ID	11	ma_4	ma_3	ma_2	ma_1	0	1	1	0
Mode A code (ma)	12	ma_{12}	ma_{11}	ma_{10}	ma_9	ma_8	ma_7	ma_6	ma_5
Call sign left (csl)	13	csl_8	csl_7	csl_6	csl_5	csl_4	csl_3	csl_2	csl_1
		csl_{I6}	csl_{I5}	csl_{I4}	csl_{I3}	csl_{I2}	csl_{11}	csl_{I0}	csl_9
	14	csr_3	csr_2	csr_1	csl_{21}	csl_{20}	csl_{I9}	csl_{I8}	csl_{I7}
Call sign right (csr)	15	csr_{11}	csr_{10}	csr_9	csr_8	csr_7	csr_6	csr_5	csr_4
	16	csr_{19}	csr_{18}	csr_{17}	csr_{16}	csr_{15}	csr_{14}	csr_{13}	csr_{12}
	17	csr_{21}	csr_{20}						

NOTE 1: The call sign field provides up to eight characters of flight ID data. While ADS-B interoperability requirements require just seven characters to be represented, the last digit will be encoded with a null, per table 5.12.
NOTE 2: This variable part is intended for use in any type of airspace when an aircraft is not equipped with a Mode A transponder.
NOTE 3: The provision of Mode A data is mandatory in US airspace for both ADS-B-NRA and ADS-B-RAD applications.

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 1: "x" denotes part of fixed data field. NOTE 2: "res" refers to bits available for the information 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 (sil)	0 to 3	Values 0, 1, 2, 3 in accordance with the four SIL categories specified in ADS-B MASPS RTCA/DO-242A [i.3], clause 2.1.2.15	
Position Navigation Accuracy Category NAC_p (nacp)	0 to 11	Values 0 through 11 in accordance with the twelve NAC_p categories specified in ADS-B MASPS RTCA/DO-242A [i.3], clause 2.1.2.13	
Velocity Navigation Accuracy Category NAC_v (nacv)	0 to 4	Values 0, 1, 2, 3, 4 in accordance with the five NAC_v categories specified in ADS-B MASPS RTCA/DO-242 A [i.3], clause 2.1.2.14	
NIC_{baro} (nicb)	Binary	0 = Barometric pressure altitude has not been cross checked against another source of pressure altitude 1 = There is more than one source of barometric pressure altitude data and cross checking of one source against another has been performed	
4-bit latitude (lat4)	-	A high-resolution component to enhance the 12-bit low-resolution encoding transmitted in the fixed part (see ETSI EN 302 842-2 [2], 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 ETSI EN 302 842-2 [2], 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	
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 $altr_8$ to $altr_1$ set to 0) Bits $altr_8$ ... $altr_1$ encode the magnitude of altitude rate of change as specified in table 5.15	

Subfield	Range	Encoding	Notes
ground speed (gs)	0 to ≥ 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 ± $\frac{1}{2}^{N+1}$ 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 second	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.	
Special position indicator (spi)	Binary	0 = No SPI 1 = SPI set (IDENT button selected)	

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.
5.1.1.5a	On receipt of the two-slot TCP or the single slot TCP variable part, local CPR decoding shall be used to decode the TCP positions, using the current position of the transmitting aircraft as the reference position.
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 (bgo) subfield shall be as defined in table 5.13.
NOTE: The longitude (lon) subfield is truncated by two bits. This means that the available resolution will be ±150 m, except at high latitudes where the resolution will decrease to ±450 m.	

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 ≤ 24 012,5	24 012,5 < base alt		base alt ≤ 24 012,5	24 012,5 < base alt
bgo < -2 075	bgo < -5 050	1	< -2 075	< -5 050
-2 075 ≤ bgo < -2 025	-5 050 ≤ bgo < -4 950	2	-2 050	-5 000
-2 025 ≤ bgo < -1 975	-4 950 ≤ bgo < -4 850	3	-2 000	-4 900
-1 975 ≤ bgo < -1 925	-4 850 ≤ bgo < -4 750	4	-1 950	-4 800
-1 925 ≤ bgo < -1 875	-4 750 ≤ bgo < -4 650	5	-1 900	-4 700
-1 875 ≤ bgo < -1 825	-4 650 ≤ bgo < -4 550	6	-1 850	-4 600
↓	↓	↓	↓	↓
3 975 ≤ bgo < 4 025	7 050 ≤ bgo < 7 150	123	4 000	7 100
4 025 ≤ bgo < 4 075	7 150 ≤ bgo < 7 250	124	4 050	7 200
4 075 ≤ bgo < 4 125	7 250 ≤ bgo < 7 350	125	4 100	7 300
bgo ≥ 4 125	bgo ≥ 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 (gs) 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 ≤ speed < 0,5	1	0
0,5 ≤ speed < 1,5	2	1
1,5 ≤ speed < 2,5	3	2
2,5 ≤ speed < 3,5	4	3
3,5 ≤ speed < 4,5	5	4
↓	↓	↓
1 021,5 ≤ speed < 1 022,5	1 023	1 022
1 022,5 ≤ speed < 1 024	1 024	1 023
1 024 ≤ speed < 1 026	1 025	1 025
1 026 ≤ speed < 1 028	1 026	1 027
↓	↓	↓
3 068 ≤ speed < 3 070	2 047	3 069
3 070 ≤ speed < 3 074	2 048	3 072
3 074 ≤ speed < 3 078	2 049	3 076
3 078 ≤ speed < 3 082	2 050	3 080
↓	↓	↓
11 250 ≤ speed < 11 254	4 093	11 252
11 254 ≤ speed < 11 258	4 094	11 256
11 258 ≤ speed	4 095	more than or equal to 11 258
NOTE: The time over which the ground speed is computed is station-dependent.		

Requirement reference	
5.1.1.9	The encoding of the magnitude part of the altitude rate (altr) subfield (bits 1 to 8) shall be as defined in table 5.15.
NOTE: The quantization step size of altitude rate (100 fpm) is equivalent to one sigma error of 0,5 fps.	

Table 5.15: Altitude rate magnitude encoding and decoding

Actual altitude rate of transmitting station (fpm)	Transmitted decimal value of $altr_8$ ... $altr_1$	Decoded magnitude of altitude rate (fpm)
unknown	0	unknown
|arate| < 50	1	0
50 ≤ |arate| < 150	2	100
150 ≤ |arate| < 250	3	200
250 ≤ |arate| < 350	4	300
↓	↓	↓
19 950 ≤ |arate| < 20 050	201	20 000
20 050 ≤ |arate| < 20 300	202	20 200
20 300 ≤ |arate| < 20 500	203	20 400
↓	↓	↓
29 900 ≤ |arate| < 30 100	251	30 000
30 100 ≤ |arate| < 30 500	252	30 250
30 500 ≤ |arate| < 31 000	253	30 750
31 000 ≤ |arate| < 32 000	254	31 500
32 000 ≤ |arate|	255	more than or equal to 32 000

Requirement reference	
5.1.1.10	The encoding of the aircraft category (ac) 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 (st) 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

Requirement reference	
	Use of information fields
5.1.1.13	Information fields 3 hex and 4 hex shall not be transmitted by mobile stations.
5.1.1.14	Information fields 0 hex, 1 hex, 2 hex, 8 hex, 9 hex, 5 hex, A1 hex, and AA0 hex shall only be transmitted by mobile stations.

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.
NOTE:	The default reporting rate is defined by the reservation data. The contingency reporting rate, in the event that certain position or velocity deviation thresholds are exceeded, is defined by the secondary reporting rate (snr) field.

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.
NOTE 1:	The value of k = 6 if the autonomous monitoring (sleep) bit = 0 (implying that the auxiliary information for sleep mode is not transmitted), and k = 8 if the sleep bit = 1.
NOTE 2:	The requested information ID (r-id) field, which is of variable length, should be encoded in the minimum length allowed. Then, for r-id selections supported within 4 bits, the parameter set enabled by the autonomous information (auto) bit fits in a single octet while the overall ADS-B request (with the autonomous monitoring (sleep) parameters and requested variable part identification) fits in a single-slot message of length = 19 octets.
NOTE 3:	As an example, an ADS-B request for an aircraft off the airport movement area, which is stopped, could potentially be sent with the encoding of table 5.22a. This encoding imposes sleep mode with wake up parameters, i.e. maximum sleep velocity (vel) = 4 knots and maximum sleep position (pos) = 10 m, and requests the high resolution variable part. The secondary reporting rate (snr) field indicates that the station should transmit once every second in the event that the velocity or position deviation thresholds are exceeded.

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	$r\text{-}id_4$	$r\text{-}id_3$	$r\text{-}id_2$	$r\text{-}id_1$
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 ETSI EN 302 842-2 [2]	n/a	
parameter group length (lg)	0 to 3	Binary	n/a	Indicates number of additional octets in parameter block (e.g. a parameter block 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 ETSI EN 302 842-2 [2], table 5.76		

Table 5.22a: Example of ADS-B request bit encoding for sleep mode with request for high precision variable part

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
requested message ID (r-mi_1) (bit 4 = 0), burst ID, Autonomous monitoring (sleep), Autonomous information (auto), requested base altitude (r-b/a)	5	sleep (1)	auto (1)	r-b/a_2	r-b/a_1	0	0	0	1
secondary reporting rate (snr), maximum sleep velocity (vel)	6	snr_4 (1)	snr_3 (1)	snr_2 (0)	snr_1 (1)	vel_4 (0)	vel_3 (1)	vel_2 (0)	vel_1 (0)
maximum sleep position (pos)	7	pos_8(0)	pos_7 (0)	pos_6 (0)	pos_5 (0)	pos_4 (1)	pos_3 (0)	pos_2 (1)	pos_1 (0)
requested information field ID (r-id)	8	lg_2 (0)	lg_1 (0)	aux_2 (0)	aux_1 (0)	r-id_4 (1)	r-id_3 (0)	r-id_2 (1)	r-id_1 (0)

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 (see note): (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.

Requirement reference	
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).
	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 5.22b.
NOTE: The procedures for sleep mode apply to vertical as well as horizontal movement.	

Table 5.22b: 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 5.22c.

Table 5.22c: 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
NOTE 1: The elements that can trigger an SVQ status change are SIL, NACP and NACV. NOTE 2: The aim is to achieve at least five reports within a twelve second period. This rate ensures that all stations have a high probability of receiving notification of an SVQ or TCP change within twelve seconds.				

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 ETSI EN 302 842-2 [2]) 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.
	Conflicting commands
5.1.4.10	In the case that a mobile station receives conflicting instructions from an ADS-B request and a channel management command, the instructions from the channel management command shall take precedence.
5.1.4.11	In the case that a mobile station receives conflicting instructions from an ADS-B request and a directed request, the instructions from the directed request shall take precedence.

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 ETSI EN 302 842-2 [2], 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 ETSI EN 302 842-2 [2], table 5.30	
directed offset (do)	See ETSI EN 302 842-2 [2], 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 ETSI EN 302 842-2 [2] 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

NOTE: The command set ID (csid) 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).

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 (see notes 1 to 3).
5.1.5.2.7	The number of transmissions per minute shall be designated NTM.

NOTE 1: A repeat cycle can be less than a submultiple of one minute, exactly one minute, or multiple minutes.
NOTE 2: The definition of the script block is provided in clause 5.1.5.2.25ff. The script consists of one or more single octet fields each referring to a single transmission and one or more two octet fields each referring to one or more transmissions (as defined in the repeat count subfield). If there are n single octet fields, m double octet fields and, for each double octet field, a number of transmissions pm, then: NES = n + sum (m pm) and sc = n + 2m.
NOTE 3: The length of a single channel management parameter is limited to 255 octets because a single octet CTRL length field is used in the CTRL parameter format.
NOTE 4: The value of NTM is defined in clause 5.1.5.2.17.

Requirement reference	
	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.
NOTE 1: If the destination block is an implied broadcast (dc = 0 per table 5.27), it is considered to match the receiving station's ID.	
NOTE 2: Bit 8 of the first octet is used to indicate the start of the addressed destination block. If set to zero, the addressed destination block is present. If set to one a regional multicast block is present (see clause 5.1.5.2.11). Hence the individual offset for the first destination in the addressed destination block is limited to 0 to 15 and not 0 to 31.	

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 ETSI EN 302 842-2 [2], 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 directed offset (do) equal to zero, or with the individual offset (ido) 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 the destination count equal to zero (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 directed offset (do) not equal to zero, and with the individual offset (ido) not 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 the 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 channel management parameter command.
5.1.5.2.15	Void.
NOTE: Directed slot operation is announced by setting bit 1 of octet 1 of the transmitted burst = 1.	

Requirement reference	
	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 ETSI EN 302 842-2 [2], 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.
NOTE 1: The (radial, range) pairs may be parsed in a clockwise direction around the defined centre of the region. Each range applies at its associated radial and in a clockwise direction until another radial is encountered. NOTE 2: A station that matches the regional broadcast filter operates according to the directed rate mechanism.	

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	$lon4_4$	$lon4_3$	$lon4_2$	$lon4_1$	$lat4_4$	$lat4_3$	$lat4_2$	$lat4_1$
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	radial 2 (r_2)	10	$r_{2,8}$	$r_{2,7}$	$r_{2,6}$	$r_{2,5}$	$r_{2,4}$	$r_{2,3}$	$r_{2,2}$	$r_{2,1}$
(as needed)	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 ETSI EN 302 842-2 [2], clause 5.6	Sent using type=0, reference position is location of station transmitting this parameter.
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 NM
1	distance < 1,0 NM
↓	↓
59	distance < 29,5 NM
60	distance < 30 NM
61	distance < 31 NM
↓	↓
89	distance < 59 NM
90	distance < 60 NM
91	distance < 62 NM
↓	↓
119	distance < 118 NM
120	distance < 120 NM
121	distance < 124 NM
↓	↓
254	distance < 656 NM
255	distance < 660 NM

Requirement reference	
	Channel management parameter reception procedures
5.1.5.2.17b	Upon receipt of a channel management parameter 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 × sr (where sr is the result of looking up the encoded sr in table 5.28 in ETSI EN 302 842-2 [2]).
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 channel management 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 mth individual offset idom shall correspond to the mth 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 5.1.5.2.17i.
1	any	Invalid
do > 1	dt < 15	Reserve the following slots for the destination to broadcast: for j equal to 0 to min(dt, 3) and k equal to 0 to NTM - 1, the slots equal to truncate (do + ido_m + (k × M1/NTM) + j × M1) through (lg + truncate (do + ido_m + (k × M1/NTM) + j × 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 × M1/NTM)) through (lg + truncate (do + (k × M1/NTM))) after the first slot of the received burst
NOTE: The station is only required to update its reservation table with the reservations that require transmission by the station. Slot selections by other stations commanded in directed rate mode are unknown at the time the command is processed. Reservations for other stations commanded in directed slot mode are known in principle, but are required to be located in blocked or quarantined slots. Hence there is no operational need to record these reservations at the time the command is processed.		

Requirement reference	
	Action on receipt of a channel management parameter with do = 0
5.1.5.2.17i	On receipt of an channel management command with the mth destination address matching that of the receiving station and with either the directed offset subfield (do) or the individual offset subfield (ido) 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 × 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 kth element identified in the script occurs truncate ((k-1)×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 channel management command with the mth destination address (d_m) matching that of the receiving station and with both the directed offset (do) and the individual offset (ido) subfields greater than 1, the responder station shall begin the transmission of the first script element at $T0_m$ slots after the first slot of the channel management command where $T0_m$ = do + ido_m.
5.1.5.2.17m	Subsequent transmission slots shall be spaced such that the slot for the kth element identified in the script occurs truncate ((k - 1) × 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 on receipt of a channel management parameter with do > 1
5.1.5.2.17o	Upon receipt of a channel management command with the directed offset subfield (do) greater than 1, the station shall set the TV11 timer (see ETSI EN 302 842-2 [2], clause 5.2.10.2) equal to the value of the directed timeout (dt) subfield for each of the slots indicated in the channel management parameter 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 timeout (pt) subfield set to min (3, TV11-1), and with the periodic offset (po) subfield set to zero if do > 1 in the received channel management parameter.
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.

Requirement reference	
	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 receiving a cancellation channel management parameter with the directed timeout (dt) 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 of a reservation made according to the instructions of a previous channel management parameter, and all other subfields set to the same values as in the previous channel management parameter, shall cancel all operations associated with the previous channel management parameter.
5.1.5.2.19	Void.
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 a station is already transmitting autonomously the same VSS user data, on the same frequencies as listed in the frequency block of a newly received channel management parameter, then it shall cancel its existing reservations on those frequencies in accordance with clause 5.2.10.5.27 in ETSI EN 302 842-2 [2], and operate in accordance with the parameters of the newly received channel management parameter.
	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 on the frequencies listed in the frequency block of the received channel management parameter with csid = 0, and initiate operations in accordance with this 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 shall terminate any previous operations commanded by the previous channel management parameter and initiate operations in accordance with the newly received channel management parameter.
5.1.5.2.19e	A station that is transmitting an accordance with an active channel management parameter, which does not receive a new channel management parameter within (dt $\times$ M1) slots, with dt as specified in the active channel management parameter, shall terminate the operations on the frequencies listed in the frequency block of the active channel management parameter and revert to default sync burst operations on those frequencies.
5.1.5.2.19f	A station that receives a channel management parameter with csid = 15 shall cancel all reservations for transmission associated with, and on frequencies indicated by, previously-received active channel management parameter(s) and revert to default sync burst operations on those frequencies and on the frequencies listed in the frequency block of the received channel management parameter with csid = 15.
	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.
3	Sync burst with baro altitude, no sleep parameters, aircraft data variable field.
4 to 15	Reserved.
NOTE: If the preferred altitude encoding is not available, then send the other if it is available.	

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 gs_{12} = 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.
NOTE: Octets 3 through to len+2 are all application-specific.	

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	$id1_4$	$id1_3$	$id1_2$	$id1_1$	$id2_4$	$id2_3$	$id2_2$	$id2_1$
NOTE 1: "aux" refers to auxiliary parameter. NOTE 2: "b/g" refers to baro/geo altitude.									

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_j	app_{j-1}	……	app_1	len_4	len_3	len_2	len_1
NOTE: Octets 2 through to len+2 are all application specific.									

. . . . . Denotes variable length field

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 ETSI EN 302 842-2 [2], 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 (as indicated by the fc field in the header block) shall be encoded as defined in table 5.38 with subfield encodings as defined in table 5.39.
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 (see note).
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.
	Reception of frequency block containing sleep bit = 1
5.1.5.2.23aa	When a channel management parameter block is received with a sleep bit set to 1 in one or more frequency blocks the station 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.
5.1.5.2.23ab	If pos is zero, then the position test shall be ignored.
5.1.5.2.23ac	If vel is zero, then the velocity test shall be ignored.
5.1.5.2.23ad	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.
	Recommendation
5.1.5.2.23a	A station with spare receivers should monitor the remaining frequencies in the list.
NOTE: The frequencies in the frequency block are ordered in priority. Thus GSC channels are always included. They are denoted as mandatory frequencies in the frequency block (rcv = 1), and are always listed before non-GSC frequencies.	

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
NOTE 1: "sleep" denotes autonomous monitoring. NOTE 2: "auto" denotes autonomous information. NOTE 3: "rcv" denotes Must Receive. NOTE 4: "res" denotes reserved bit.									

Table 5.39: Frequency block field encoding

Subfield	Encoding	Notes
frequency (f)	See ETSI EN 302 842-2 [2], 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	Per clause 5.1.2.2	If the sleep bit is set, then the octets defined in table 5.20 are appended per clause 5.1.2.2.
auto	auto = 0: the receiving station shall choose its transmissions autonomously and ignore any reference to this frequency in the script block auto = 1: the receiving station shall transmit on this frequency according to the definition in the script block	If the auto bit is set to 0, the script block shall not refer to this frequency.

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 ETSI EN 302 842-2 [2], 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.
5.1.5.2.24d	Reception of these parameters shall replace any default settings for the same parameters, any settings resulting from a previous ground-initiated modification parameter command, and any settings resulting from a previous channel management command.
NOTE: Transmit parameter count (pc) would be greater than the length of table 5.40 if more parameters were added after the software was released.	

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	$Q2c_8$	$Q2c_7$	$Q2c_6$	$Q2c_5$	$Q2c_4$	$Q2c_3$	$Q2c_2$	$Q2c_1$
Q2d	4	$Q2d_8$	$Q2d_7$	$Q2d_6$	$Q2d_5$	$Q2d_4$	$Q2d_3$	$Q2d_2$	$Q2d_1$
Q2a, Q2b, Q2c, Q2d	5	$Q2d_{10}$	$Q2d_9$	$Q2c_{10}$	$Q2c_9$	$Q2b_{10}$	$Q2b_9$	$Q2a_{10}$	$Q2a_9$
Q1, Q4	6	$Q1_3$	$Q1_2$	$Q1_1$	$Q4_5$	$Q4_4$	$Q4_3$	$Q4_2$	$Q4_1$
Q1, V12	7	$Q1_4$	$V12_7$	$V12_6$	$V12_5$	$V12_4$	$V12_3$	$V12_2$	$V12_1$
VS2 (db)	8	res	res	$VS2_6$	$VS2_5$	$VS2_4$	$VS2_3$	$VS2_2$	$VS2_1$
VS4 (NM)	9	res	$VS4_7$	$VS4_6$	$VS4_5$	$VS4_4$	$VS4_3$	$VS4_2$	$VS4_1$
NOTE: "res" refers to bits available for the information field.									

Requirement reference	Script block definition
5.1.5.2.25	The channel management 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 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 *bits* 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 lat4, lon4 (for *bits* = 3), lat6, lon6 (for *bits* = 5) or lat8, lon8 (for *bits* = 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: *clat*$_{in}$ = latitude to be encoded. *clon*$_{in}$ = latitude to be encoded. *lat* = the CPR encoded latitude. *lon* = the CPR encoded longitude. *type* = type of CPR (odd or even). *bits* = 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: $diff(pos_1, pos_2) = \begin{cases} pos_1 - pos_2 - MAX_C - 1 & \text{if } pos_1 > pos_2 + MAX_C / 2 \\ pos_1 - pos_2 + MAX_C + 1 & \text{if } pos_2 > pos_1 + MAX_C / 2 \\ pos_1 - pos_2 & \text{else} \end{cases}$ $sign(pos_1, pos_2) = \begin{cases} 1 & \text{if } diff(pos_1, pos_2) \geq 0 \\ 0 & \text{else} \end{cases}$

Requirement reference	
	Latitude
5.1.6.1.5	The encoded fixed latitude offset shall be calculated as follows: $lat = enc_{lat}(clat_{in}, type)$ $clat_{dec} = dec_{lat}(clat_{in}, lat, lat, type)$ $tmp_7 = \frac{MAX_C}{2 \cdot nz(type) \cdot MAX_T^{lat} \cdot (2^{bits} - 1)}$ $offset_{enc}^{lat}(clat_{in}, clat_{dec}, bits, type) = \frac{\lvert diff(clat_{in}, clat_{dec}) \rvert + \frac{tmp_7}{2}}{tmp_7}$ $lat4(bits\ 1to3) = offset_{enc}^{lat}(clat_{in}, clat_{dec}, 3, type)$ $lat6(bits\ 1to5) = offset_{enc}^{lat}(clat_{in}, clat_{dec}, 5, type)$ $lat8(bits\ 1to7) = offset_{enc}^{lat}(clat_{in}, clat_{dec}, 7, type)$ $lat4(bit\ 4) = lat6(bit\ 6) = lat8(bit\ 8) = sign(clat_{in}, clat_{dec})$
	Longitude
5.1.6.1.6	The encoded fixed longitude offset shall be calculated as follows: $lon = enc_{lon}(clat_{dec}, clon_{in}, type)$ $clon_{dec} = dec_{lon}(clat_{dec}, clon_{in}, lon, lon, type)$ $tmp_8 = \frac{MAX_C}{2 \cdot nl(clat_{dec}, type) \cdot MAX_T^{lon} \cdot (2^{bits} - 1)}$ $offset_{enc}^{lon}(clat_{dec}, clon_{in}, clon_{dec}, bits, type) = \frac{\lvert diff(clon_{in}, clon_{dec}) \rvert + \frac{tmp_8}{2}}{tmp_8}$ $lon4(bits1to3) = offset_{enc}^{lon}(clat_{dec}, clon_{in}, clon_{dec}, 3, type)$ $lon6(bits1to5) = offset_{enc}^{lon}(clat_{dec}, clon_{in}, clon_{dec}, 5, type)$ $lon8(bits1to7) = offset_{enc}^{lon}(clat_{dec}, clon_{in}, clon_{dec}, 7, type)$ $lon4(bit\ 4) = lon6(bit\ 6) = lon8(bit\ 8) = sign(clon_{in}, clon_{dec})$

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: *lat4, lat6, lat8* = received latitude offset. *lon4, lon6, lon8* = received longitude offset. *type* = type of encoding (odd or even). *clatdec* = the target's decoded latitude
	Calculations
	Latitude
5.1.6.2.3	The decoded fixed latitude offset shall be calculated as follows: $bits = \begin{cases} 3 & if\ lat4, lon4\ provided\ as\ input \\ 5 & if\ lat6, lon6\ provided\ as\ input \\ 7 & if\ lat8, lon8\ provided\ as\ input \end{cases}$ $lat_{offs} = \begin{cases} lat4(bits1to3) & if\ lat4, lon4\ provided\ as\ input \\ lat6(bits1to5) & if\ lat6, lon6\ provided\ as\ input \\ lat8(bits1to7) & if\ lat8, lon8\ provided\ as\ input \end{cases}$ $s_{lat} = \begin{cases} lat4(bit4) & if\ lat4, lon4\ provided\ as\ input \\ lat6(bit6) & if\ lat6, lon6\ provided\ as\ input \\ lat8(bit8) & if\ lat8, lon8\ provided\ as\ input \end{cases}$ $tmp_9 = \frac{MAX_C}{2 \cdot nz(type) \cdot MAX_T^{lat} \cdot (2^{bits} - 1)}$ $offset_{dec}^{lat}(lat_{offs}, s_{lat}, bits, type) = \begin{cases} lat_{offs} \cdot tmp_7 & \text{if } s_{lat} = 1 \\ -lat_{offs} \cdot tmp_7 & \text{if } s_{lat} = 0 \end{cases}$
	Longitude
5.1.6.2.4	The decoded fixed longitude offset shall be calculated as follows: $lon_{offs} = \begin{cases} lon4(bits1to3) & if\ lat4, lon4\ provided\ as\ input \\ lon6(bits1to5) & if\ lat6, lon6\ provided\ as\ input \\ lon8(bits1to7) & if\ lat8, lon8\ provided\ as\ input \end{cases}$ $s_{lon} = \begin{cases} lon4(bit4) & if\ lat4, lon4\ provided\ as\ input \\ lon6(bit6) & if\ lat6, lon6\ provided\ as\ input \\ lon8(bit8) & if\ lat8, lon8\ provided\ as\ input \end{cases}$ $tmp_{10} = \frac{MAX_C}{2 \cdot nl(clat_{dec}, type) \cdot MAX_T^{lon} \cdot (2^{bits} - 1)}$ $offset_{dec}^{lon}(clat_{dec}, lon_{offs}, s_{lon}, bits, type) = \begin{cases} lon_{offs} \cdot tmp_8 & \text{if } s_{lon} = 1 \\ -lon_{offs} \cdot tmp_8 & \text{if } s_{lon} = 0 \end{cases}$
NOTE 1: tmp_7 is defined in clause 5.1.6.1.5. NOTE 2: tmp_8 is defined in clause 5.1.6.1.6.	

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: $lat_p = \begin{cases} \frac{clat_{in}}{dlat(type)} & \text{if } 0 \le clat_{in} \le \frac{MAX_C}{4} \\ \frac{clat_{in}}{dlat(type)} - 16 & \text{if } clat_{in} > \frac{MAX_C}{4} \end{cases}$ $lat = enc_{lat}(clat_{in}, type)$ $clat_{dec} = dec_{lat}(clat_{in}, lat, lat, type)$ $lon_p = \frac{clon_{in}}{dlon(clat_{dec}, type)}$ $pid = enc_{patch}(lat_p, lon_p) = 36 \cdot lat_p + lon_p$ $offset_{dec}^{lat}(lat_{offs}, s_{lat}, bits, type) = \begin{cases} lat_{offs} \cdot tmp_7 & \text{if } s_{lat} = 1 \\ -lat_{offs} \cdot tmp_7 & \text{if } s_{lat} = 0 \end{cases}$

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: *lat* = CPR latitude to decode. *lon* = CPR longitude to be decode. *type* = type of encoding (odd or even). *pid* = encoded patch id.
	Calculations
	Latitude
5.1.6.4.3	The decoded latitude component of the patch ID shall be calculated as follows: $lat_p = \begin{cases} \frac{pid}{36} & \text{if } 0 \le pid \le 359 \\ \frac{pid}{36} + 16 & \text{if } pid > 359 \end{cases}$ $fullDec_{lat}(lat, lat_p, type) = \frac{\left(\left(\frac{MAX_C}{MAX_T^{lat}}\right) \cdot lat\right)}{nz(type)} + dlat(type) \cdot lat_p$

	Longitude
5.1.6.4.4	The patch ID longitude component shall be calculated as follows: $lon_p = \mathrm{mod}(pid, 36)$ $clat_{dec} = fullDec_{lat}(lat, lat_p, type)$ $fullDec_{lon}(clat_{dec}, lon, lon_p, type) = \frac{\left(\left(\frac{MAX_C}{MAX_T^{lon}}\right) \cdot lon\right)}{nl(clat_{dec}, type)} + dlon(clat_{dec}, type) \cdot lon_p$

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 ETSI EN 302 842-2 [2].
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 ETSI EN 302 842-2 [2]).
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.

Requirement reference	
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 + tmi_1.
5.2.3.8	A station shall be capable of decoding TIS-B bursts appended with the following reservation fields: - a reservation ID (rid) equal to 1, with the burst appended by a null reservation field as defined in clause 5.9 of ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2], 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; 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 ETSI EN 302 842-2 [2].

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	$vlat1_8$	$vlat1_7$	$vlat1_6$	$vlat1_5$	$vlat1_4$	$vlat1_3$	$vlat1_2$	$vlat1_1$
vertex longitude (vlg(1 -k))	n + 10	$vlg1_8$	$vlg1_7$	$vlg1_6$	$vlg1_5$	$vlg1_4$	$vlg1_3$	$vlg1_2$	$vlg1_1$
		.....							
	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 ETSI EN 302 842-2 [2]	
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

Data field	Definition	Encoded Value
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 ETSI EN 302 842-2 [2]	
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

Data field	Definition	Encoded Value
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)	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_1	$rlat_{20}$	$rlat_{19}$	$rlat_{18}$	$rlat_{17}$	$rlat_{16}$	$rlat_{15}$	$rlat_{14}$
	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 ETSI EN 302 842-2 [2]	
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 5.11)	See table 5.4.7	
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 corr(lat) as defined in table 5.52, where lat = RND(latitude) is the vertex or target latitude rounded down to the nearest half degree.

Table 5.52: Values for scaling function corr(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 = Dlat × 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 = Dlat × 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 = Dlat × 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 degree: Encoded vertex latitude = Dlat × 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 = Dlat × 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	Dlon = vertex longitude - reference longitude; Dlon is in degrees 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. For Dlon (converted to NM) between -240 NM and -64 NM: Encoded vertex longitude = INT(Dlon × corr(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). For Dlon (converted to NM) between -60 NM and -21 NM: Encoded vertex longitude = INT(Dlon × corr(vertex latitude) × 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). For Dlon (converted to NM) between -20 NM and +20 NM: Encoded vertex longitude = INT(Dlon × corr(vertex latitude) × 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). For Dlon (converted to NM) between +21 NM and +60 NM: Encoded vertex longitude = INT(Dlon × corr(vertex latitude) × 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 degree at equator; range is between +15,75 and +45 degrees near the poles). For Dlon (converted to NM) between +64 NM and +240 NM: Encoded vertex longitude = INT(Dlon × corr(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). Unused	0 to 44 45 to 84 85 to 165 166 to 205 206 to 250 251 to 255
Airborne target latitude	Dtlat = target latitude - reference latitude; Dtlat is in degrees For Dtlat between -4 and +4 degrees: Encoded target latitude = Dtlat × 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) Unused	0 to 65 400 65 401 to 65 536
Airborne target longitude	Dtlon = target latitude - reference latitude; Dtlon is in degrees For Dtlon (converted to NM) between -240 NM and +240 NM: Encoded target longitude = INT(Dtlon × corr(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). Unused	0 to 65 400 65 401 to 65 536

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 × 130 800 + 523 200 (Dglat has an increment of 1/130 800 of a degree; target resolution ±0,45 m; range from -240 NM to +240 NM) Unused	0 to 1 046 400 1 046 401 to 1 048 575
Ground target longitude	Dglon = target latitude - reference latitude; Dglon is in degrees For Dglon (converted to NM) between -240 NM and +240 NM: Encoded target longitude = INT(Dglon × corr(target latitude) × 16 + 523 200) (Dglon has an increment of 1/130 800 of a degree at the equator; target resolution ±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). Unused	0 to 1 046 400 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 ETSI EN 302 842-2 [2].
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 ETSI EN 302 842-2 [2]).
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	A station will be capable of decoding FIS-B bursts appended with the following reservation fields: - a reservation ID (rid) equal to 1, with the burst appended by a null reservation field as defined in clause 5.2.9 of ETSI EN 302 842-2 [2]; - a reservation ID (rid) equal to 0 and an extended reservation ID (erid) equal to 00000 binary, with the burst appended by a response reservation field as defined in clause 5.2.18 of ETSI EN 302 842-2 [2], 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; 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 ETSI EN 302 842-2 [2].

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 1 to15	
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 ≤ time < 1 1 ≤ time < 2 2 ≤ time < 3 3 ≤ time < 4 ↓ 1 438 ≤ time < 1 439 1 439 ≤ 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 ≤ heading < 5 5 ≤ heading < 15 15 ≤ heading < 25 25 ≤ heading < 35 ↓ 315 ≤ heading < 325 325 ≤ heading < 335 335 ≤ heading < 345 345 ≤ 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 < speed < 0,5 0,5 ≤ speed < 1,5 1,5 ≤ speed < 2,5 2,5 ≤ speed < 3,5 ↓ 96,5 ≤ speed < 97,5 97,5 ≤ speed < 98,5 98,5 ≤ speed < 99,5 99,5 ≤ speed < 100,5 100,5 ≤ 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	0	
	Not present	1	
Variable wind direction to (degrees)	Unknown	0	
	355 ≤ variable < 5	1	0
	5 ≤ variable < 15	2	10
	15 ≤ variable < 25	3	20
	25 ≤ variable < 35	4	30
	↓	↓	↓
	315 ≤ variable < 325	33	320
	325 ≤ variable < 335	34	330
	335 ≤ variable < 345	35	340
	345 ≤ variable < 355	36	350
	Unused	37 to 63	
Variable wind direction from (degrees)	As above		
Wind Gust Indicator	As for wind speed		
Minimum visibility (metres)	Unknown	0	
	0 ≤ visibility < 100	1	50
	100 ≤ visibility < 150	2	100
	150 ≤ visibility < 200	3	150
	200 ≤ visibility < 250	4	200
	↓	↓	↓
	700 ≤ visibility < 750	14	700
	750 ≤ visibility < 800	15	750
	800 ≤ visibility < 900	16	800
	900 ≤ visibility < 1 000	17	900
	↓	↓	↓
	6 000 ≤ visibility < 7 000	41	6 000
	7 000 ≤ visibility < 8 000	42	7 000
	8 000 ≤ visibility < 9 000	43	8 000
	9 000 ≤ visibility < 10 000	44	9 000
	10 000 ≤ visibility < 11 000	45	10 000
	Unused	46 to 63	
Minimum visibility direction	North	0	0
	North East	1	1
	East	2	2
	South East	3	3
	South	4	4
	South West	5	5
	West	6	6
	North West	7	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	0	
	2	1	
	3	2	
	4	3	

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

Data field	Definition	Encoded value	Decoded value (if different)
State of the sea	Unknown 0 1 2 3 4 5 6 7 8 9 Unused	0 1 2 3 4 5 6 7 8 9 10 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 [4] 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
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

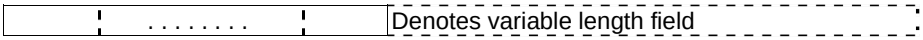

Table 5.59: ATIS message field encoding

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 5 min 10 min 15 min 20 min 25 min 30 min 35 min 40 min 45 min 50 min 55 min 60 min Unused Unused Unknown Delay	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	
Transition level	Unknown 0 5 10 15 ↓ 285 290 295 300 Reserved for future use	0 1 2 3 4 ↓ 58 59 60 61 62 to 63	
Other operational activity flag	Present Not present	0 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 Closed	 1 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, 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	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_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
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 ≤ length < 50	1	25
	50 ≤ length < 100	2	75
	100 ≤ length < 150	3	125
	150 ≤ length < 200	4	175
	↓	↓	
	6 300 ≤ length < 6 350	127	6 325
	6 350 ≤ length < 6 400	128	6 375
	6 400 ≤ length < 6 450	129	6 425
	6 450 ≤ length	130	
	Unused	131 to 254	
	Full length	255	
Reduced runway width (metres)	Not reported	0	
	0 ≤ width < 1	1	1
	1 ≤ width < 2	2	2
	2 ≤ width < 3	3	3
	3 ≤ width < 4	4	4
	↓	↓	
	67 ≤ width < 68	68	68
	68 ≤ width < 69	69	69
	69 ≤ width < 70	70	70
	70 ≤ 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	

5.3.6 SIGnificant METeorological Information (SIGMET) message

Requirement reference	
5.3.6.1	A station will be capable of decoding a SIGMET message as defined in table 5.62.
5.3.6.2	Void.
5.3.6.3	Void.
5.3.6.4	Fields within the SIGMET message shall be decoded as defined in table 5.63.

NOTE: The information contained within the free text portion of the SIGMET message will include the following information detailed in abbreviated plain language using approved ICAO abbreviations:

Application FIR: the name of the flight information region or control area for which the SIGMET message is issued.

Type of Information: this will detail whether the weather phenomenon is an observed or forecast condition. If the weather condition is an observed condition the time of observation will be indicated in UTC.

Location and Level: this will give an indication, referring where possible to latitude and longitude and/or locations or geographic features known well internationally.

Movement and Speed: the movement or expected movement of the phenomenon will be given with reference to one of the eight points of the compass and given in terms of kilometres per hour, knots per hour, or stationary.

Changes in Intensity: the free text will indicate the changes in intensity of the weather condition expressing this in terms of intensifying, weakening or no change.

Additional Information: additional information will detail any further pertinent information to the weather phenomenon. This may also include an outlook providing information beyond the period for which the SIGMET is valid.

Table 5.62: SIGMET 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 sequence number (sqn)	6	sqn_4	sqn_3	sqn_2	sqn_1	fmi_4	fmi_3	fmi_2	fmi_1
message edition (med)	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)	12	di_5	di_4	di_3	di_2	di_1	ori_{20}	ori_{19}	ori_{18}
time issued (ti)	13	ti_8	ti_7	ti_6	ti_5	ti_4	ti_3	ti_2	ti_1
date valid to (dvt)	14	dvt_5	dvt_4	dvt_3	dvt_2	dvt_1	ti_{11}	ti_{10}	ti_9
time valid to (tvt)	15	tvt_8	tvt_7	tvt_6	tvt_5	tvt_4	tvt_3	tvt_2	tvt_1
SIGMET flag (sfl) free text flag (ftxt) phenomenon / condition reported (crp)	14	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

. . . . . Denotes variable length field

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 Message	0 1
SIGMET flag	Normal Convective SIGMET	0 1
Phenomenon / Condition reported	Condition reported in free text Thunderstorm Thunderstorm obscured Thunderstorm embedded Thunderstorm frequent Thunderstorm squall line Thunderstorm obscured with heavy hail Thunderstorm embedded with heavy hail Thunderstorm frequent with heavy hail Thunderstorm squall line with heavy hail Tropical cyclone Freezing rain Moderate turbulence Severe turbulence Severe icing Severe icing due to freezing rain Severe mountain waves Heavy dust storm Heavy sand storm Volcanic ash Isolated cumulonimbus Occasional cumulonimbus Frequent cumulonimbus Reserved for future use	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 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)	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	$freq_{11}$	$freq_{10}$	$freq_9$	$freq_8$	$freq_7$	$freq_6$	$freq_5$	$freq_4$
…… 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	$actim_3$	$actim_2$	$actim_1$	$acdat_5$	$acdat_4$	$acdat_3$	$acdat_2$	$acdat_1$
	15	$actim_{11}$	$actim_{10}$	$actim_9$	$actim_8$	$actim_7$	$actim_6$	$actim_5$	$actim_4$
…… 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 TSAs	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 corresponds 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 ETSI EN 302 842-2 [2], 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: - 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 ETSI EN 302 842-2 [2].
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 ETSI EN 302 842-2 [2]).
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: - a reservation ID (rid) equal to 1, with the burst appended by a null reservation field as defined in clause 5.9 of ETSI EN 302 842-2 [2]; - a reservation ID (rid) equal to 0 and an extended reservation ID (erid) equal to 00000 binary, with the burst appended by a response reservation field as defined in clause 5.2.18 of ETSI EN 302 842-2 [2], 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; - 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 ETSI EN 302 842-2 [2]; 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 ETSI EN 302 842-2 [2].

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) measurement type (t)	14	t_3	t_2	t_1	n_5	n_4	n_3	n_2	n_1
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". (See note).	
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 minute 00,0 s 00 minute 00,1 s 00 minute 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 x 10^{-6} m/m 10 x 10^{-6} m/m ..... 1,275 x 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 second 10 s 20 s ….. 2 530 s 2 540 s ≤ 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	0
	-32,766 m	1
	-32,765 m	2
	…..	…..
	0 m	32 767
	0,001 m	32 768
	…..	…..
	+32,766 m	65 533
	+32,767 m	65 534
	Unused	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
	0,02 m	1
	0,04 m	2
	…..	…..
	5,06 m	253
	5,08 m	254
	Ranging source correction invalid	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 [i.4] and GBAS MOPS [i.5], clause A.2.1).	
	Bit 8:	
	Positive	0
	Negative	1
	Bits 1 to 7:	
	0 m	0
	0,05 m	1
	…..	…..
	6,34 m	126
	6,35 m	127
	Reference receiver was not used to compute the pseudorange correction:	1000 0000
GNS-B Message CRC (gc)	See clause 5.4.5.1	
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.		

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

Data field	Definition	Encoded Value
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	0
	GCID 1	1
	GCID 2	2
	GCID 3	3
	GCID 4	4
	Reserved	5
	Reserved	6
	GNS-B unhealthy	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)	0
	Negative, denotes west variation (counter-clockwise from true north)	1
	Bits 2 to 11:	
	0 degree	0
	0,25 degree	1
	0,50 degree	2
	…..	…..
	180 degrees	720
	Unused	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
	0,1 x 10^{-6} m/m	1
	0,2 x 10^{-6} m/m	2
	…..	…..
	25,5 x 10^{-6} m/m	255
Refractivity index (i)	This defines the refractivity index parameter value stored in the GNS-B Ground Subsystem.	
	Bit 8:	
	Positive	0
	Negative	1
	Bits 1 to 7:	
	0	0
	3	1
	….	…..
	378	126
	381	127
Scale height (h)	This defines the scale height parameter value stored in the GNS-B Ground Subsystem.	
	0	0
	100	1
	200	2
	…..	…..
	25 400	254
	25 500	255
Refractivity uncertainty (u)	This defines the refractivity uncertainty parameter value stored in the GNS-B Ground Subsystem.	
	0	0
	1	1
	2	2
	…..	…..
	255	255

Data field	Definition	Encoded Value
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 degree 0 minute 0 arcsecond 0 degree 0 minute 0,0005 arcsecond 0 degree 0 minute 0,00010 arcsecond ….. 89 degrees 59 minutes 59,9995 arcseconds 90 degrees 0 minute 0 arcsecond 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 degree 0 minute 0 arcsecond 0 degree 0 minute 0,0005 arcsecond 0 degree 0 minute 0,00010 arcsecond ….. 89 degrees 59 minutes 59,9995 arcseconds 90 degrees 0 minute 0 arcsecond ….. 180 degrees 0 minute 0 arcsecond 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
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 [i.4] and clause A.2.4 in GBAS MOPS [i.5] 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
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

Description	Octet	Bit number							
		8	7	6	5	4	3	2	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 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. (See note). 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

Data field	Definition	Encoded Value
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
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 (ltla)	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 degree 0 minute 0 arcsecond 0 degree 0 minute 0,0005 arcsecond 0 degree 0 minute 0,00010 arcsecond ….. 89 degrees 59 minutes 59,9995 arcseconds 90 degrees 0 minute 0 arcsecond Unused	 0 1 0 1 2 ….. 647 999 999 648 000 000 648 000 001 to 2 147 483 647

Data field	Definition	Encoded Value
LTP/FTP longitude (ltlo)	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 degree 0 minute 0 arcsecond 0 degree 0 minute 0,0005 arcsecond 0 degree 0 minute 0,00010 arcsecond ….. 89 degrees 59 minutes 59,9995 arcseconds 90 degrees 0 minute 0 arcsecond ….. 180 degrees 0 minute 0 arcsecond 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
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 degree 0 minute 0 arcsecond 0 degree 0 minute 0,0005 arcsecond ….. 1 degree 0 minute 0 arcsecond 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 degree 0 minute 0 arcsecond 0 degree 0 minute 0,0005 arcsecond ….. 1 degree 0 minute 0 arcsecond Unused	 0 1 0 1 ….. 7 200 000 7 200 001 to 8 388 607

Data field	Definition	Encoded Value
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
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 degree 0,01 degree 0,02 degree ….. 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	 0 1 2 3 ….. 254 255
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	

Data field	Definition	Encoded Value
FAS vertical alert limit/approach status (va)	Vertical Alert Limit	
	0 m	0
	0,1 m	1
	0,2 m	2
	…..	….
	25,4 m	254
	Do not use vertical deviations.	255
FAS lateral alert limit/approach status (la)	Lateral Alert Limit	
	0 m	0
	0,2 m	1
	0,4 m	2
	…..	…..
	50,8 m	254
	Do not use approach	255
GNS-B Message CRC (gc)	See clause 5.4.5.1	
NOTE: This parameter is not used for approaches conducted using the GNS-B Ground Subsystem corrections.		

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: $M(x) = \sum_{i=1}^{n} m_i x^{n-i} = m_1 x^{n-1} + m_2 x^{n-2} + ... + m_n x^0$
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: $M(x) = \sum_{i=1}^{272} m_i x^{272-i} = m_1 x^{271} + m_2 x^{270} + ... + m_{272} x^0$
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: $M(x) = \sum_{i=1}^{n} m_i x^{n-i} = m_1 x^{n-1} + m_2 x^{n-2} + ... + m_n x^0$
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 ETSI EN 302 842-2 [2], clause 6.

7 Protocol test procedures

7.1 General

The equipment shall meet the requirements of ETSI EN 302 842-2 [2], 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 ETSI EN 302 842-2 [2], clause 7.2.

7.3 Protocol test-suite description methodology

The equipment shall meet the requirements of ETSI EN 302 842-2 [2], 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 parameter 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 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 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 parameter 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 parameter if it is in the correct region (according to Condition 1), and that processing of the channel management parameter will terminate if it is not.
Channel_Multicast_C	To demonstrate that a mobile station will act on a regional multicast channel management parameter if it is in the correct region (according to Condition 2a), and that processing of the channel management parameter 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 parameter if it is in the correct region (according to Condition 2b), and that processing of the channel management parameter 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 parameter will be judged invalid, and the entire parameter 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 parameter, then the message will be judged invalid and the entire parameter 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 parameter, 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 parameter, it shall use NTM as the default reporting rate for the frequency to construct the plea response.
Channel_Cancel_DT	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.
Channel_Cancel_CSID_A	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.
Channel_Cancel_CSID_B	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.
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 channel management command.
Channel_Cancel_CSID_C	To demonstrate that a station that receives a channel management parameter with csid = 0 will terminate any previous default sync burst operations and initiate operations in accordance with the current channel management parameter.
Channel_Cancel_CSID_D	To demonstrate that 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.
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 parameter, 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 parameter 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 parameter 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 parameter 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 parameter will transmit all of the streams defined in the script while applying the V12 parameter.

Test Case Name	Description
Channel_Param_VS4	To demonstrate that a station in receipt of a channel management parameter 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 parameter 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 command containing a script block with two consecutive octets having bits 6 to 8 equal to "111".
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.

If a station will implement the TIS-B functionality then it shall conform to the tests with prefix "TIS_". Otherwise, it shall not be required to conform to these tests.

If a station will implement the FIS-B functionality then it shall conform to the tests with prefix "FIS_". Otherwise, it shall not be required to conform to these tests.

If a station will implement the GNS-B functionality then it shall conform to the tests with prefix "GNS_". Otherwise, it shall not be required to conform to these tests.

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 ETSI EN 302 842-2 [2], 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
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

Mnemonic	Meaning
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
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

Mnemonic	Meaning
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
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
rrl	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

Mnemonic	Meaning
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
tvt	Time valid to
txd	Transmission definition index
txt	Free text message
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) and other simulated stations are assumed to be at 0° latitude and at 0° longitude, unless otherwise specified. The positions of other stations are given in terms of the direction (East, E, is used for all cases) and distance in nautical miles with respect to the position of station A.

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

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 [i.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 target is assumed to become unreachable during the elapsed time represented by the missed reports (gaps in the table), due to expiration of L1.

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	±0,0012	4	±0,0003
Decoded lon	14	±0,0012 (see note 1)	5	±0,0003
Decoded 4-bit lat	16	±0,000076	6	±0,00002
Decoded 4-bit lon	18	±0,000076 (see note 1)	6	±0,00002
Decoded 6-bit lat	18	±0,000019	6	±0,000005
Decoded 6-bit lon	20	±0,000019 (see note 1)	7	±0,000005
Decoded 8-bit lat	20	±0,0000048	7	±0,0000012
Decoded 8-bit lon	22	±0,0000048 (see note 1)	7	±0,0000012
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 sut	latitude of the station under test
lon sut	longitude of the station under test
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 [i.6], 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. The position of the station under test is varied, since this shall remain within plus or minus 300,5 NM of the received position for local decode to be carried out (which uses the position of the station under test as a reference).

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

lat sut	lon sut	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
12,9	-0,8	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
12,9	-0,8	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	12,872009	-0,7986998
12,9	-0,8	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	12,888302	-0,7824004
12,9	-0,8	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	12,904605	-0,7661002
12,9	-0,7	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	12,920901	-0,7498003
13,0	-0,7	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	12,953496	-0,7172001
13,0	-0,7	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	12,969801	-0,7009008
13,0	-0,7	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	13,002395	-0,6683011
13,1	-0,6	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	13,051301	-0,6193998
13,1	-0,6	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	13,100196	-0,5704995
13,2	-0,5	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	13,165403	-0,5053002
13,3	-0,4	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
13,3	-0,4	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
13,3	-0,3	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	13,328402	-0,3422992
13,4	-0,3	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	13,409007	-0,2608009
13,4	-0,2	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	13,426203	-0,2445002
13,4	-0,2	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	13,442502	-0,2282008
13,5	-0,2	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	13,475104	-0,1956007
13,5	-0,2	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	13,491400	-0,1793008
13,5	-0,2	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	13,507696	-0,1630006
13,5	-0,1	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	13,524004	-0,1467004
13,5	-0,1	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	13,540297	-0,1303987
13,6	-0,1	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	13,556597	-0,1141012
13,6	-0,1	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	13,589201	-0,0814994

lat sut	lon sut	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
13,7	0,0	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	13,654398 7	-0,0163009
13,7	0,1	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	13,735898 4	0,0651994
13,9	0,2	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	13,850000 6	0,1792996
13,9	0,2	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	13,866295 6	0,1956006
14,0	0,3	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	13,996702 3	0,3259994
14,0	0,3	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	14,012998 9	0,3423010
14,1	0,5	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
14,2	0,5	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
14,3	0,6	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	14,290098 4	0,6193991
14,3	0,7	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	14,322700 0	0,6519987
14,4	0,7	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	14,404202 8	0,7334995
14,4	0,8	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	14,436796 5	0,7660987
14,5	0,8	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	14,518300 6	0,8475994
14,6	0,9	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
14,6	1,0	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
14,7	1,0	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
14,7	1,1	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	14,746502 8	1,0757997
14,8	1,1	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	14,762803 0	1,0920988
14,9	1,2	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
14,9	1,2	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	14,893197 7	1,2225009
15,0	1,3	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
15,1	1,4	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	15,088799 6	1,4180988
15,1	1,5	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
15,2	1,5	15,154000000	15,121400000	1,483300000	1,450700000	1	1	0	2	n	20	y	2j	L1	3	15,1535	1,48358	15,154070	1,483292	15,154018	1,4832985	15,153997 0	1,4832999
15,2	1,6	15,235500000	15,154000000	1,564800000	1,483300000	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

lat sut	lon sut	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
15,3	1,6	15,268100000	15,235500000	1,597400000	1,564800000	0	0	0	2	n	20	y	2b	L1	3	15,2674	1,59764	15,268097	1,597408	15,268108	1,5973991	15,268101 1	1,5973997
15,3	1,7	15,349600000	15,268100000	1,678900000	1,597400000	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
15,4	1,7	15,365900000	15,349600000	1,695200000	1,678900000	0	1	0	2	n	10	n	2f	GL	4	15,3651	1,69523	15,365952	1,695183	15,365906	1,6951980	15,365896 6	1,6951987
15,5	1,8	15,480000000	15,365900000	1,809300000	1,695200000	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
15,5	1,8	15,496300000	15,480000000	1,825600000	1,809300000	0	1	0	2	n	10	y	2c	GL	4	15,4969	1,82578	15,496250	1,825596	15,496317	1,8256035	15,496303 3	1,8256000
15,6	1,9	15,610400000	15,496300000	1,939700000	1,825600000	1	0	0	4	n	70	y	4p	L1	3	15,6107	1,93970	15,610316	1,939705	15,610391	1,9397047	15,610398 2	1,9396994
15,6	2,0	15,626700000	15,610400000	1,956000000	1,939700000	0	1	0	3	n	10	n	3f	GL	4	15,6264	1,95569	15,626722	1,956009	15,626689	1,9559985	15,626700 5	1,9559987
15,7	2,1	15,740800000	15,626700000	2,070100000	1,956000000	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
15,8	2,1	15,757100000	15,740800000	2,086400000	2,070100000	0	1	0	2	y	10	n	2a	GP	4	15,7582	2,08624	15,757021	2,086422	15,757100	2,0864040	15,757097 7	2,0864000
15,9	2,2	15,871200000	15,757100000	2,200500000	2,086400000	1	0	0	4	n	70	y	4p	L1	3	15,8719	2,20073	15,871182	2,200491	15,871211	2,2005036	15,871197 4	2,2005011
15,9	2,2	15,887500000	15,871200000	2,216800000	2,200500000	0	1	0	3	y	10	n	3a	GP	4	15,8877	2,21679	15,887493	2,216789	15,887510	2,2167990	15,887504 4	2,2167987
15,9	2,2	15,903800000	15,887500000	2,233100000	2,216800000	1	0	0	4	y	10	n	4a	GP	4	15,9046	2,23336	15,903835	2,233119	15,903783	2,2330994	15,903801 0	2,2331003
15,9	2,2	15,920100000	15,903800000	2,249400000	2,233100000	0	1	0	4	n	10	n	4i	GL	4	15,9194	2,24910	15,920112	2,249380	15,920084	2,2493951	15,920096 5	2,2494010
19,1	5,4	19,098600000	-	5,427900000	-	1	-	-	1	n	-	n	1e	NO	2	NO CALC	NO CALC	NO CALC	NO CALC	NO CALC	NO CALC	NO CALC	NO CALC
19,1	5,4	19,114900000	19,098600000	5,444200000	5,427900000	0	1	0	2	n	10	n	2f	GL	4	19,1160	5,44438	19,114949	5,444196	19,114892	5,4442038	19,114899 1	5,4442004
19,1	5,5	19,131200000	19,114900000	5,460500000	5,444200000	1	0	0	4	n	10	n	4q	GL	4	19,1322	5,46021	19,131285	5,460491	19,131210	5,4604954	19,131203 1	5,4604990
19,1	5,5	19,147500000	19,131200000	5,476800000	5,460500000	0	1	0	4	n	10	n	4i	GL	4	19,1477	5,47670	19,147567	5,476788	19,147505	5,4768000	19,147500 8	5,4768000
19,2	5,5	19,163800000	19,147500000	5,493100000	5,476800000	1	0	1	4	n	10	n	4zb	L2	4	19,1648	5,49281	19,163759	5,493107	19,163782	5,4931006	19,163796 8	5,4930992
19,2	5,5	19,180100000	19,163800000	5,509400000	5,493100000	0	1	0	4	n	10	n	4i	GL	4	19,1795	5,50948	19,180185	5,509385	19,180117	5,5094053	19,180102 5	5,5093992
23,5	9,8	23,483300000	-	9,812600000	-	1	-	-	1	n	-	y	1d	L1	3	23,4826	9,81276	23,483292	9,812616	23,483304	9,8125971	23,483296 5	9,8126010
23,5	9,8	23,499600000	23,483300000	9,828900000	9,812600000	0	1	0	3	n	10	y	3c	GL	4	23,4994	9,82904	23,499564	9,828893	23,499586	9,8288960	23,499601 0	9,8288993
23,5	9,8	23,515900000	23,499600000	9,845200000	9,828900000	1	0	0	4	n	10	y	4l	GL	4	23,5152	9,84504	23,515945	9,845185	23,515916	9,8452037	23,515900 1	9,8451997
23,5	9,9	23,532200000	23,515900000	9,861500000	9,845200000	0	1	1	4	n	10	y	4v	L2	4	23,5311	9,86152	23,532182	9,861518	23,532199	9,8614959	23,532202 7	9,8614991
23,5	9,9	23,548500000	23,532200000	9,877800000	9,861500000	1	0	0	4	n	10	y	4l	GL	4	23,5479	9,87767	23,548419	9,877818	23,548488	9,8778035	23,548503 7	9,8778002

lat sut	lon sut	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
27,2	13,5	27,167100000	-	13,496400000	-	0	-	-	1	n	-	y	1b	L1	3	27,1673	13,49615	27,167103	13,496400	27,167120	13,4963982	27,167041	13,4964006
27,2	13,5	27,183400000	27,167100000	13,512700000	13,496400000	1	0	0	3	n	10	y	3h	GL	4	27,1824	13,51259	27,183315	13,512692	27,183390	13,5127051	27,183396	13,5126996
27,2	13,5	27,199700000	27,183400000	13,529000000	13,512700000	0	1	0	4	n	10	n	4i	GL	4	27,1990	13,52912	27,199721	13,529017	27,199693	13,5290048	27,199696	13,5289993
27,2	13,5	27,216000000	27,199700000	13,545300000	13,529000000	1	0	0	4	n	10	n	4q	GL	4	27,2151	13,54520	27,215968	13,545299	27,216002	13,5453002	27,216000	13,5453006
27,2	13,6	27,232300000	27,216000000	13,561600000	13,545300000	0	1	1	4	n	10	n	4x	L2	4	27,2332	13,56150	27,232339	13,561602	27,232305	13,5616035	27,232297	13,5616011
27,2	13,6	27,248600000	27,232300000	13,577900000	13,561600000	1	0	0	4	n	10	n	4q	GL	4	27,2477	13,57773	27,248621	13,577890	27,248615	13,5778984	27,248603	13,5779003
30,4	16,7	30,361900000	-	16,691200000	-	0	-	-	1	n	-	y	1b	L1	3	30,3614	16,69102	30,361940	16,691175	30,361889	16,6911946	30,361897	16,6911990
30,4	16,7	30,378200000	30,361900000	16,707500000	16,691200000	1	0	0	3	n	10	y	3h	GL	4	30,3774	16,70756	30,378112	16,707510	30,378204	16,7075036	30,378204	16,7074993
30,4	16,7	30,394500000	30,378200000	16,723800000	16,707500000	0	1	0	4	n	10	y	4d	GL	4	30,3956	16,72363	30,394558	16,723782	30,394502	16,7238012	30,394498	16,7238000
30,5	16,8	30,508600000	30,394500000	16,837900000	16,723800000	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
30,5	16,9	30,524900000	30,508600000	16,854200000	16,837900000	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
30,5	16,9	30,541200000	30,524900000	16,870500000	16,854200000	1	0	0	2	n	10	n	2k	GL	4	30,5407	16,87073	30,541198	16,870509	30,541186	16,8705060	30,541203	16,8704992
30,6	16,9	30,557500000	30,541200000	16,886800000	16,870500000	0	1	0	4	n	10	n	4i	GL	4	30,5568	16,88702	30,557474	16,886808	30,557486	16,8868044	30,557497	16,8868008
33,4	19,7	33,361100000	-	19,690400000	-	0	-	-	1	y	-	y	1a	GP	4	33,3602	19,69017	33,361068	19,690376	33,361101	19,6904029	33,361099	19,6904002
33,4	19,7	33,377400000	33,361100000	19,706700000	19,690400000	1	0	0	4	n	10	y	4l	GL	4	33,3765	19,70689	33,377349	19,706725	33,377384	19,7067038	33,377401	19,7066991
33,4	19,7	33,393700000	33,377400000	19,723000000	19,706700000	0	1	0	4	n	10	y	4d	GL	4	33,3944	19,72313	33,393686	19,723023	33,393714	19,7229977	33,393700	19,7230007
33,5	19,8	33,507800000	33,393700000	19,837100000	19,723000000	1	0	0	4	n	70	y	4p	L1	3	33,5071	19,83722	33,507782	19,837107	33,507794	19,8371056	33,507796	19,8370993
33,5	19,9	33,524100000	33,507800000	19,853400000	19,837100000	0	1	1	3	n	10	y	3n	L1	3	33,5238	19,85313	33,524158	19,853398	33,524085	19,8533967	33,524097	19,8533993
33,5	19,9	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	33,540399	19,8696998
33,6	19,9	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	33,556996	19,8859993
36,1	22,5	36,132100000	-	22,461400000	-	1	-	-	1	y	-	y	1a	GP	4	36,1319	22,46135	36,132048	22,461405	36,132111	22,4613999	36,132095	22,4613987
36,1	22,5	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	36,148404	22,4776990
36,2	22,5	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	36,164692	22,4940013

lat sut	lon sut	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
36,3	22,6	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	22,608102	36,278017	22,608099
36,3	22,6	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
36,3	22,6	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	22,640698	36,311403	22,640706
36,3	22,7	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	22,657002	36,327697	22,656998
38,7	25,1	38,723800000	-	25,053100000	-	1	-	-	1	y	-	n	1a	GP	4	38,7240	25,05280	38,723839	25,053089	38,723816	25,053100	38,723801	25,053099
38,7	25,1	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	25,069394	38,740099	25,069398
38,8	25,1	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	25,085694	38,756404	25,085698
38,9	25,2	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
38,9	25,2	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	25,216102	38,886799	25,216100
38,9	25,2	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	25,232397	38,903098	25,232400
38,9	25,2	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	25,248695	38,919403	25,248698
41,2	27,5	41,185100000	-	27,514400000	-	0	-	-	1	n	-	y	1b	L1	3	41,1844	27,51470	41,185069	27,514410	41,185120	27,514398	41,185101	27,514399
41,2	27,5	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	27,530706	41,201399	27,530700
41,2	27,5	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	27,547004	41,217036	27,547001
41,3	27,7	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	27,661105	41,331803	27,661100
41,3	27,7	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
41,4	27,7	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	27,693701	41,364397	27,693700
41,4	27,7	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	27,710001	41,380702	27,709999
43,5	29,9	43,532300000	-	29,861600000	-	0	-	-	1	y	-	y	1a	GP	4	43,5311	29,86182	43,532357	29,861576	43,532317	29,861599	43,532298	29,861601
43,5	29,9	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	29,877902	43,548602	29,877899
43,6	29,9	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	29,894205	43,564005	29,894199
43,7	30,0	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
43,7	30,0	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	30,024603	43,695297	30,024598

lat sut	lon sut	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
43,7	30,0	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	30,040905 9	43,711600 9	30,040898 5
43,7	30,1	43,727900000	43,711600000	30,057200000	30,040900000	0	1	0	4	n	10	y	4d	GL	4	43,7289	30,0569 1	43,727891	30,057227	43,727914	30,057196 2	43,727899 4	30,057199 9
45,8	32,1	45,781700000	-	32,111000000	-	1	-	-	1	y	-	y	1a	GP	4	45,7821	32,1113 3	45,781745	32,111008	45,781698	32,110995 3	45,781698 2	32,110999 7
45,8	32,1	45,798000000	45,781700000	32,127300000	32,111000000	0	1	0	4	n	10	y	4d	GL	4	45,7973	32,1268 6	45,798012	32,127303	45,797983	32,127302 7	45,797996 4	32,127299 2
45,8	32,1	45,814300000	45,798000000	32,143600000	32,127300000	1	0	0	4	n	10	y	4l	GL	4	45,8148	32,1433 8	45,814218	32,143577	45,814311	32,143601 8	45,814301 8	32,143600 2
45,9	32,3	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
45,9	32,3	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
46,0	32,3	45,961000000	45,944700000	32,290300000	32,274000000	0	1	0	2	n	10	n	2f	GL	4	45,9609	32,2898 7	45,960928	32,290331	45,961007	32,290301 9	45,960995 3	32,290299 0
46,0	32,3	45,977300000	45,961000000	32,306600000	32,290300000	1	0	0	4	n	10	n	4q	GL	4	45,9780	32,3065 5	45,977304	32,306621	45,977293	32,306599 4	45,977300 1	32,306598 3
47,9	34,3	47,949600000	-	34,278900000	-	1	-	-	1	y	-	n	1a	GP	4	47,9498	34,2784 8	47,949585	34,278890	47,949602	34,278896 7	47,949596 4	34,278898 1
48,0	34,3	47,965900000	47,949600000	34,295200000	34,278900000	0	1	0	4	n	10	n	4i	GL	4	47,9658	34,2950 0	47,965986	34,295197	47,965891	34,295192 9	47,965898 5	34,295199 2
48,0	34,3	47,982200000	47,965900000	34,311500000	34,295200000	1	0	0	4	n	10	n	4q	GL	4	47,9824	34,3119 2	47,982238	34,311510	47,982215	34,311503 2	47,982200 0	34,311501 8
48,1	34,4	48,096300000	47,982200000	34,425600000	34,311500000	0	1	0	4	n	70	y	4f	L1	3	48,0952	34,4259 3	48,096285	34,425602	48,096302	34,425604 4	48,096295 7	34,425601 3
48,1	34,4	48,112600000	48,096300000	34,441900000	34,425600000	1	0	1	3	n	10	y	3p	L1	3	48,1130	34,4422 4	48,112671	34,441887	48,112584	34,441905 0	48,112604 6	34,441901 2
48,1	34,5	48,128900000	48,112600000	34,458200000	34,441900000	0	1	0	3	n	10	y	3c	GL	4	48,1294	34,4580 9	48,128903	34,458231	48,128914	34,458202 0	48,128897 3	34,458199 5
48,1	34,5	48,145200000	48,128900000	34,474500000	34,458200000	1	0	0	4	n	10	y	4l	GL	4	48,1457	34,4742 1	48,145145	34,474491	48,145197	34,474495 5	48,145198 3	34,474500 5

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 degree (north)	359 degrees	0 degree (north)	290	0	2 042	0
Assumed base altitude (balt) to calculate baro/geo offset (bgo) from (ft)	20 000 feet	20 000 feet	30 000 feet	30 000 feet	1 413	1 413	1 813	1 813

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_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
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	0	Climbing	Climbing	Descending	Climbing
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 degree (north)	359 degrees	0 degree (north)
Assumed base altitude (balt) to calculate baro/geo offset (bgo) from (ft)	1 413	1 413	1 813	1 813	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	Climbing	0	0	1	0
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 degree (north)	359 degrees	0 degree (north)	580	0	4 085	0
Assumed base altitude (balt) to calculate baro/geo offset (bgo) from (ft)	20 000 feet	20 000 feet	30 000 feet	30 000 feet	1 413	1 413	1 813	1 813

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_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
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	0	0	0	1	Barometric altitude rate	Barometric altitude rate	Barometric altitude rate	Geometric altitude rate
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	0	0	1	0	Climbing	Climbing	Descending	Climbing
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	580	0	4 085	0	51 degrees	0 degree (north)	359 degrees	0 degree (north)
Assumed base altitude (balt) to calculate baro/geo offset (bgo) from (ft)	1 413	1 413	1 813	1 813	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_Encode	As encoded in test CPR_Encode	As encoded in test CPR_Encode	As encoded in test CPR_Encode
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
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
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 degree (north)	359 degrees	0 degree (north)	290	0	2 042	0
Assumed base altitude (balt) to calculate baro/geo offset (bgo) from (ft)	20 000 feet	20 000 feet	30 000 feet	30 000 feet	1 413	1 413	1 813	1 813

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_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode
pid	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode
lon6	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode	As decoded in test CPR_Decode
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 degree (north)	359 degrees	0 degree (north)
Assumed base altitude (balt) to calculate baro/geo offset (bgo) from (ft)	1 413	1 413	1 813	1 813	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_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)**
h	5	1	23	0	5am	1am	11pm	midnight
min	27	1	59	0	27 minutes	1 minute	59 minutes	0 minute
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
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_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 second	60 seconds	0 second
Assumed base altitude (balt) to calculate baro/geo offset (bgo) from (ft)	1 413	1 413	1 813	1 813	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
mon	6	1	12	1	June	January	December	January
min	27	1	59	0	27 minutes	1 minute	59 minutes	0 minute
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 second	60 seconds	0 second

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	934	2	4 071	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 minute	0 minute	15,25 minutes	unknown	3	1	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	934	2	4 071	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	0,75 minute	0 minute	15,25 minutes	unknown	4	1	62	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)**
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	934	2	4 071	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 minute	0 minute	15,25 minutes	unknown	5	1	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	934	2	4 071	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 minute	15,25 minutes	unknown	6	1	62	0
csl	AAAA	BBBB	CCCC	DDDD	0x000000	0x00CB5C	0x0196B8	0x026214
st	no emergency/not reported	general emergency	no communications	unlawful interference	0	1	4	5
csr	0000	1111	2222	3333	0x14A758	0x1572B4	0x163E10	0x17096C
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
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

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)**
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	Climbing	0	0	1	0
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 degree (north)	359 degrees	0 degree (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_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
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	934	2	4 071	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	1	62	0	0,5 minute	0 minute	15,25 minutes	unknown
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	934	2	4 071	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	1	62	0	0,75 minute	0 minute	15,25 minutes	unknown
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	934	2	4 071	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	1	62	0	1,0 minute	0 minute	15,25 minutes	unknown
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	934	2	4 071	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	1	62	0	1,25 minutes	0 minute	15,25 minutes	unknown
csl	0x000000	0x00CB5C	0x0196B8	0x026214	AAAA	BBBB	CCCC	DDDD
st	0	1	4	5	no emergency/not reported	general emergency	no communications	unlawful interference
csr	0x14A758	0x1572B4	0x163E10	0x17096C	0000	1111	2222	3333

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)**
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	0	Climbing	Climbing	Descending	Climbing
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
gs	1 023	1	2 047	0	1 022	0	3 069	unknown
gt	290	0	2 042	0	51 degrees	0 degree (north)	359 degrees	0 degree (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	934	2	4 071	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 minute	0 minute	15,25 minutes	unknown	4	1	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_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
balt	934	2	4 071	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	1	62	0	0,75 minute	0 minute	15,25 minutes	unknown

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 < 555,6 m (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	0x000000	0x00CB5C	0x0196B8	0x026214
csr	0000	1111	2222	3333	0x14A758	0x1572B4	0x163E10	0x17096C

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	0x000000	0x00CB5C	0x0196B8	0x026214	AAAA	BBBB	CCCC	DDDD
csr	0x14A758	0x1572B4	0x163E10	0x17096C	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	51 degrees	0 degree (north)	359 degrees	0 degree (north)	580	0	4 085	0
tind	Unknown	Left	Right	Straight	0	1	2	3

The values in 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_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
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	580	0	4 085	0	51 degrees	0 degree (north)	359 degrees	0 degree (north)
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 second	60,0 seconds	27 seconds
Accuracy of TIS-B targets (NM unless otherwise stated)	6	9	1	0	< 0,1 NM	< 3m	< 10 NM	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 degree 41 minutes west	0 degree 41 minutes west	0 degree 41 minutes west	0 degree 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 degree 41 minutes	-0 degree 41 minutes	NOT TRANSMITTED	-0 degree 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 degree 5 minutes	0 degree 5 minutes	NOT TRANSMITTED	0 degree 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 ADS-B as a data source	TIS-B target based only on radar or multilat. data	TIS-B target includes ADS-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 degree	359,70 degrees	0,00 degree
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 ADS-B as a data source	TIS-B target based only on radar or multilat. data	TIS-B target includes ADS-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 degree	359,70 degrees	0,00 degree
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_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	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 ADS-B as a data source	TIS-B target based only on radar or multilat. data	TIS-B target includes ADS-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 degree	359,70 degrees	0,00 degree
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_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	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 minute	1 439 minutes	0 minute
Wdi	15	1	36	1	140 degrees	0 degree	350 degrees	0 minute
Wsp	46	1	102	1	45 kts	0 kt	100,5 kts	0 kt
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 degree	350 degrees	0 degree
Vdif	16	1	36	1	150 degrees	0 degree	350 degrees	0 degree
Wgi	47	1	102	1	46,0 kts	0,0 kt	100,5 kts	0,0 kt
Mis	16	1	45		800 meters	0 meter	10 000 meters	NOT TRANSMITTED
misd	0	1	5		North	North East	South West	NOT TRANSMITTED
Mas	18	1	45		1 000 meters	0 meter	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 meter	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$ TEST FREE TEXT		TREND MESSAGE $	TEST FREE TEXT	TREND MESSGAE TEST FREE TEXT	NOT TRANSMITTED TEST FREE TEXT	TREND MESSAGE TEST FREE TEXT	NOT TRANSMITTED 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 minute
Hd	6	0	12	0	30 minutes	0 minute	60 minutes	0 minute
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 degree	350 degrees	0 degree
Wsp	46	1	102	1	45 kts	0 kt	100,5 kts	0 kt
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 degree	350 degrees	Unknown
Vdif	16	1	36	0	150 degrees	0 degree	350 degrees	Unknown
Wgi	47	1	102	1	46 kts	0,1 kt	100,5 kts	0 kt
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
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

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**
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) Txt (FREE TEXT)	TREND MESSGAE $ TEST FREE TEXT		TREND MESSGAE $	TEST FREE TEXT	TREND MESSGAE TEST FREE TEXT	NOT TRANSMITTED NOT TRANSMITTED	TREND MESSGAE 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 minute
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 meter	Full length	0 meter
rrw	33	1	127	1	32 meters	0 meter	Full width	0 meter
rdp	1	1	0	0	Negative	Negative	Positive	Positive
rvt	23	1	32		700 meters	0 meter	1 500 meters	NOT TRANSMITTED
rvm	22	1	32		650 meters	0 meter	1 527 meters	NOT TRANSMITTED
rve	24	1	32		750 meters	0 meter	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 minute	1 439 minutes	0 minute
Di	14	1	31	0	day 14	day 1	day 31	unknown
Ti	721	2	1 440	1	720 minutes	1 minute	1 439 minutes	0 minute
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 *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	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 minute	1 439 minutes	0 minute
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 *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	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 minute	1 439 minutes	0 minute
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_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	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 second	00 minute 00,0 second	20 minutes 00,0 second	00 minute 00,0 second
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
ad	1	0	255	0	10 seconds	0 second	Prediction of source availability duration not provided	0 second
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

GNS_TYPE1_RF_IN_PARAMETERS(x)					GNS_TYPE1_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)**
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_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	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 degree	0 degree	180,00 degrees	0 degree
vi	1	0	255	0	$0,1 \times 10^{-6}$	0	$25,5 \times 10^{-6}$	0

GNS_TYPE2_RF_IN_PARAMETERS(x)					GNS_TYPE2_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)**
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 degree 0 minute 0,0005 arcsecond	0 degree 0 minute 0 arcsecond	90 degrees 0 minute 0 arcsecond	0 degree 0 minute 0 arcsecond
lon (bit 32)	1	0	1	0	Positive	Negative	Positive	Negative
lon (bits 1 to 31)	1	0	1 296 000 000	0	0 degree 0 minute 0,0005 arcsecond	0 degree 0 minute 0 arcsecond	180 degrees 0 minute 0 arcsecond	0 degree 0 minute 0 arcsecond
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

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)**
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 degree 0 minute 0,0005 arcsecond	0 degree 0 minute 0 arcsecond	90 degrees 0 minute 0 arcsecond	0 degree 0 minute 0 arcsecond
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 degree 0 minute 0,0005 arc-second	0 degree 0 minute 0 arcsecond	180 degrees 0 minute 0 arcsecond	0 degree 0 minute 0 arcsecond
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 degree 0 minute 0,0005 arcsecond	0 degree 0 minute 0 arcsecond	1 degree 0 minute 0 arcsecond	0 degree 0 minute 0 arcsecond
ΔFPAP Longitude (bit 24)	1	0	1	0	Positive	Negative	Positive	Negative
ΔFPAP Longitude (bits 1 to 23)	1	0	1 200 000	0	0 degree 0 minute 0,0005 arcsecond	0 degree 0 minute 0 arcsecond	1 degree 0 minute 0 arcsecond	0 degree 0 minute 0 arcsecond
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 degree	0 degree	90 degrees	0 degree
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 ETSI EN 302 842-2 [2]. This clause defines additional bursts and frames not already defined in ETSI EN 302 842-2 [2].

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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.42: 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
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	$lon4_4$	$lon4_3$	$lon4_2$	$lon4_1$	$lat4_4$	$lat4_3$	$lat4_2$	$lat4_1$
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.43: 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	$lat6_6$	$lat6_5$	$lat6_4$	$lat6_3$	$lat6_2$	$lat6_1$
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	$lon6_6$	$lon6_5$	$lon6_4$	$lon6_3$	$lon6_2$	$lon6_1$
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.44: 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	$lon4_4$	$lon4_3$	$lon4_2$	$lon4_1$	$lat4_4$	$lat4_3$	$lat4_2$	$lat4_1$
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.45: 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	$lon4_4$	$lon4_3$	$lon4_2$	$lon4_1$	$lat4_4$	$lat4_3$	$lat4_2$	$lat4_1$
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.46: 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}
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, nucr, 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	53	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.47: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.48: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.49: 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
po	19	po_8	po_7	po_6	po_5	po_4	po_3	po_2	po_1
c	20	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.50: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.51: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	18	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.52: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.53: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	11	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.54: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.55: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	14	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.56: 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	$r\text{-}id_4$	$r\text{-}id_3$	$r\text{-}id_2$	$r\text{-}id_1$
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	15	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.57: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	19	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

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	$r\text{-}id_4$	$r\text{-}id_3$	$r\text{-}id_2$	$r\text{-}id_1$
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	20	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.59: 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	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
dt	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
trmt	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, nr; pr_flag = 0	12	or	$rcvr_2$	$rcvr_1$	0	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	16	0	1	1	0	0	d_{27}	d_{26}	d_{25}
c	17	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	18	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.60: 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	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	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	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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.61: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	53	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.62: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	18	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.63: 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
c	21	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	22	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.64: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	23	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.65: 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	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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	26	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.66: 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	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
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	$lon4_4$	$lon4_3$	$lon4_2$	$lon4_1$	$lat4_4$	$lat4_3$	$lat4_2$	$lat4_1$
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	$d_{1,8}$	$d_{1,7}$	$d_{1,6}$	$d_{1,5}$	$d_{1,4}$	$d_{1,3}$	$d_{1,2}$	$d_{1,1}$

Description	Octet	Bit number							
		8	7	6	5	4	3	2	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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	29	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.67: 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
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	23	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.68: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	20	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.69: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.70: 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
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	22	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.71: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	23	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.72: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	20	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.73: 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

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
erid, d	19	0	0	0	0	0	1	1	1
c	20	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.74: 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	$TV11min_4$	$TV11min_3$	$TV11min_2$	$TV11min_1$	$TV11max_4$	$TV11max_3$	$TV11max_2$	$TV11max_1$
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	33	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.75: 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	$lat6_6$	$lat6_5$	$lat6_4$	$lat6_3$	$lat6_2$	$lat6_1$
in	13	x	x	$lon6_6$	$lon6_5$	$lon6_4$	$lon6_3$	$lon6_2$	$lon6_1$
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.76: 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	$lon4_4$	$lon4_3$	$lon4_2$	$lon4_1$	$lat4_4$	$lat4_3$	$lat4_2$	$lat4_1$
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.77: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.77a: SYNC_BURST_j (Sj): Occupies one slot - Autonomous burst, high dynamic variable part - Lat4 and lon4 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	1
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	$lon4_8$	$lon4_7$	$lon4_6$	$lon4_5$	$lat4_4$	$lat4_3$	$lat4_2$	$lat4_1$
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

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	$vlat1_8$	$vlat1_7$	$vlat1_6$	$vlat1_5$	$vlat1_4$	$vlat1_3$	$vlat1_2$	$vlat1_1$
vertex longitude (vlg(1 -k))	16	$vlg1_8$	$vlg1_7$	$vlg1_6$	$vlg1_5$	$vlg1_4$	$vlg1_3$	$vlg1_2$	$vlg1_1$
		.....							
	m-4	$vlatk_8$	$vlatk_7$	$vlatk_6$	$vlatk_5$	$vlatk_4$	$vlatk_3$	$vlatk_2$	$vlatk_1$
	m-3	$vlgk_8$	$vlgk_7$	$vlgk_6$	$vlgk_5$	$vlgk_4$	$vlgk_3$	$vlgk_2$	$vlgk_1$
response reservation	m-2	0	0	0	0	0	1	1	1
c	m-1	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	m	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.79: 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)	7	tid_5	tid_4	tid_3	tid_2	tid_1	fidflg	adsff	rflg
target identifier (tid)	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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	27	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.80: 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)	7	tid_5	tid_4	tid_3	tid_2	tid_1	fidflg	adsff	rflg
target identifier (tid)	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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	27	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.81: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	19	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

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
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

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	n	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.83: 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
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	$misd_3$	$misd_2$	$misd_1$

Description	Octet	Bit number							
		8	7	6	5	4	3	2	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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	n	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.84: 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
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

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	n	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.85: 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
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	24	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.86: 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)	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)	12	di_5	di_4	di_3	di_2	di_1	ori_{20}	ori_{19}	ori_{18}
time issued (ti)	13	ti_8	ti_7	ti_6	ti_5	ti_4	ti_3	ti_2	ti_1
date valid to (dvt)	14	dvt_5	dvt_4	dvt_3	dvt_2	dvt_1	ti_{11}	ti_{10}	ti_9
time valid to (tvt)	15	tvt_8	tvt_7	tvt_6	tvt_5	tvt_4	tvt_3	tvt_2	tvt_1
SIGMET flag (sfl) free text flag (ftxt) phenomenon / condition reported (crp)	14	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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	n	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.87: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	n	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.88: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	n	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.89: 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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	13	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.90: 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	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	$locl_{20}$	$locl_{19}$	$locl_{18}$	$locl_{17}$	$locl_{16}$	$locl_{15}$	$locl_{14}$
d	10	d_{24}	d_{23}	d_{22}	d_{21}	d_{20}	d_{19}	d_{18}	d_{17}
d	11	d_{16}	d_{15}	d_{14}	d_{13}	d_{12}	d_{11}	d_{10}	d_9
d	12	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1
ro, sdf, d	13	ro_{12}	ro_{11}	ro_{10}	ro_9	sdf	d_{27}	d_{26}	d_{25}
ro	14	ro_8	ro_7	ro_6	ro_5	ro_4	ro_3	ro_2	ro_1
lg	15	lg_8	lg_7	lg_6	lg_5	lg_4	lg_3	lg_2	lg_1
pr	16	0	0	1	0	pr_4	pr_3	pr_2	pr_1
c	17	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	18	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

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	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	0	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) measurement type (t)	14	t_3	t_2	t_1	n_5	n_4	n_3	n_2	n_1
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

Description	Octet	Bit number							
		8	7	6	5	4	3	2	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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	36+(N-1)x11	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.92: 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
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	36	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.93: 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
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}

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
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_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	59+(i-1)x40	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

7.4.3.2 Test cases

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

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

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 shall be terminated and the equipment initialized before a new test case is executed. Further verification in that test case may not provide any valid results.

7.4.3.2.1 Test case macros

Test case macros shall be as defined in ETSI EN 302 842-2 [2], 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	AppIn	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	AppIn	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	AppIn	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	AppIn	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	Wait for a sync burst from station A.

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	AppIn	REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (s:=add_A; d:=add_B; r-id:=4)		Instruct station A via the AppIn PCO to send a request burst to station B requesting UTC time information field.
	8	send	AppIn	REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (s:=add_A; d:=add_C; r-id:=4)		Instruct station A via the AppIn PCO to send a request burst to station C requesting UTC time information field.
	9	send	AppIn	REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (s:=add_A; d:=add_D; r-id:=4)		Instruct station A via the AppIn PCO to send a request burst to station D requesting UTC time information field.
	10	send	AppIn	REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (s:=add_A; d:=add_E; r-id:=4)		Instruct station A via the AppIn PCO to send a request burst to station E requesting UTC time information field.
	11	send	AppIn	REQUEST TO TRANSMIT ADS-B REQUEST MESSAGE (s:=add_A; d:=add_F; r-id:=4)		Instruct station A via the AppIn 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	AppIn	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	AppIn	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	AppIn	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	AppIn	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	AppIn	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	AppIn	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		Start loop.

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	AppIn	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	AppIn	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:	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**
	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:	ADSB_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:		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**
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.
	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_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_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		Go to next parameter set.
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_PARAMETE RS(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.
	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_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_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_PARAMETERS(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_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_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_PARAMETERS(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_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_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_PARAMETERS(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_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_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 channel management 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.

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**
	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:= 4; 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, 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 channel management 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.

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**
	9	verify	RF	IF x = 2 THEN SYNC BURST = SYNC_FULL_POSITION_a and b/g = 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 flag set appropriately.
	10	rep 29		n:=1		Start loop.
	11	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.
	12	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.
	13	verify	RF	IF do_val > 1 THEN po = 0, pt = min (3, TV11 - 1) in SYNC BURST (s = add_A)		Verify that if do is greater than 1 then 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 x		Go to next value of txd.
	24	endrep		next do_val, ido_val		Go to next values of do and ido.
postamble	25					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:= 4; 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, 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 channel management 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	rep 5		n:=1		Start loop.
	10	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.
	11	verify	RF	IF n = 2 or 4 THEN SYNC BURST = SYNC_FULL_POSITION_a and b/g = 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 flag set appropriately.
	12	verify	RF	IF do_val > 1 THEN po = 0, pt = min (3, TV11 - 1) in SYNC BURST (s = add_A)		Verify that if do is greater than 1 then the sync burst contains po = 0 and pt = min (3, TV11 - 1).
	13	endrep		n:= n + 1		End loop.
	14	await		SYNC BURST (s = add_A)		Wait for a sync burst from the station under test.
	15	verify		SYNC BURST (s = add_A) = SYNC_BURST_c (s = add_A)	Sc	Verify the transmitted burst is now an autonomous sync burst.
	16	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.
	17	rep 60		n:= 1		Start loop.
	18	await		SYNC_BURST_c (s = add_A)	Sc	Wait for a sync burst from the station under test.
	19	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.
	20	endrep		n:= n + 1		End loop.

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**
	21	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.
	22	endrep		next do_val, ido_val		Go to next values of do and ido.
postamble	23					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.
	6	send	RF	CTRL_CHAN_d (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 8; 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:= 4; 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 channel management 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.
	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.

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**
	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:= 4; 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:= 11; 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 parameter if it is in the correct region (according to Condition 1), and that processing of the channel management parameter 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.
	8	send	RF	CTRL_CHAN_e (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 11; 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:= 11; 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.

Test Case Name:				Channel_Multicast_B		
Purpose:	**To demonstrate that a mobile station will act on a regional multicast channel management parameter if it is in the correct region (according to Condition 1), and that processing of the channel management parameter will terminate if it is not.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	16	do		ctime:= time when station A receives the transmission		Record the time that the station under test receives the channel management 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_Multicast_C		
Purpose:	**To demonstrate that a mobile station will act on a regional multicast channel management parameter if it is in the correct region (according to Condition 2a), and that processing of the channel management parameter 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.

Test Case Name:				Channel_Multicast_C		
Purpose:	**To demonstrate that a mobile station will act on a regional multicast channel management parameter if it is in the correct region (according to Condition 2a), and that processing of the channel management parameter 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:= 11; 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:= 11; 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 channel management 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.

Test Case Name:	Channel_Multicast_C					
Purpose:	**To demonstrate that a mobile station will act on a regional multicast channel management parameter if it is in the correct region (according to Condition 2a), and that processing of the channel management parameter will terminate if it is not.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	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 parameter if it is in the correct region (according to Condition 2b), and that processing of the channel management parameter 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:= 11; 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.

Test Case Name:				Channel_Multicast_D		
Purpose:	**To demonstrate that a mobile station will act on a regional multicast channel management parameter if it is in the correct region (according to Condition 2b), and that processing of the channel management parameter will terminate if it is not.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	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.
	15	send	RF	CTRL_CHAN_e (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 11; 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 channel management 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 parameter 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 (GSC 1)	CTRL_CHAN_b (s:= add_G; HEADER: dt:= 11; do:= 0; dc:= 4; 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 31		n:= 1		Start loop.
	4	await	RF (GSC 1)	SYNC BURST (s = add_A)	Sb	Wait for a sync burst from the station under test.
	5	record	RF (GSC 1)	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(31) - 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 GSC channel (i.e. 6 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 parameter, then the message will be judged invalid and the entire parameter 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 (GSC 1)	CTRL_CHAN_f (s:= add_G; HEADER: dt:= 6; do:= 0; dc:= 4; csid:= 1; tc:= 0; fc:= 2; styp:= 1; sr:= 0; pc:= 0; sc:= 1; 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 31		n:= 1		Start loop.
	4	await	RF (GSC 1)	SYNC BURST (s = add_A)	Sb	Wait for a sync burst from the station under test.
	5	record	RF (GSC 1)	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(31) - 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 GSC channel (i.e. 6 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 parameter, 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:= 1; 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 parameter, 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:= 4; 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.

Test Case Name:				Channel_Plea		
Purpose:	**To demonstrate that if a station receives a plea request while transmitting according to a channel management parameter, 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**
	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:= 4; 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.
	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.

Test Case Name:	Channel_Plea					
Purpose:	**To demonstrate that if a station receives a plea request while transmitting according to a channel management parameter, 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**
	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 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:= 4; 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 channel management 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		

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 parameter to be cancelled, will revert to default sync burst operations.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	8	send	RF	CTRL_CHAN_b (s:= add_G; HEADER: dt:= 15; do:= 2 250; dc:= 4; csid:= 1; tc:= 0; fc:= 1; styp:= 0; sr:= 1; pc:= 0; sc:= 0; DEST: ido:= 1; 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, with do pointing to the next slot in which the station under test was due to transmit according to the original channel management parameter and with ido set to a value greater than 0.
	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					Bring test equipment into idle state.
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:= 4; 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, 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 channel management 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:= 4; 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					Bring test equipment into idle state.
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:= 4; 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, 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 channel management 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:= 4; 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					Bring test equipment into idle state.
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 channel management 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:= 4; 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 channel management 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 parameter with csid = 0 will terminate any previous default sync burst operations and initiate operations in accordance with the current channel management 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:= 4; 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 parameter with csid = 0 will terminate any previous default sync burst operations and initiate operations in accordance with the current channel management 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 parameter with csid matching a previously-received channel management CTRL parameter will terminate any previous operations commanded by the previous channel management parameter and initiate operations in accordance with the current channel management parameter.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	rep 3		m:= {0, 1, 14}		
	3	send	RF	CTRL_CHAN_a (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 0; csid:= m; 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.
	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 (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 flag set appropriately.
	7	await		time:= sync_time + 60		Wait for the next sync burst.
	8	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.
	9	await		time:= sync_time + 65		Wait until 5 seconds after the last sync burst.
	10	send	RF	CTRL_CHAN_b (s:= add_G; HEADER: dt:= 5; do:= 100; dc:= 4; csid:= m; 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.
	11	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.
	12	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.
	13	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.
	14	verify	RF	a/d = 1 in SYNC BURST (s = add_A)		Verify that the autonomous/directed flag is set to 1.
	15	verify	RF	po = 0, pt = 3 in SYNC BURST (s = add_A)		Verify that the sync burst contains po = 0 and pt = 3.
	16	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.
	17	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 parameter with csid matching a previously-received channel management CTRL parameter will terminate any previous operations commanded by the previous channel management parameter and initiate operations in accordance with the current channel management parameter.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	18	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.
	19	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.
	20	rep 29		n:=1		Start loop to check the rest of the sync bursts commanded by the second channel management command.
	21	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.
	22	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.
	23	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).
	24	verify	RF	a/d = 1 in SYNC BURST (s = add_A)		Verify that the autonomous/directed flag is set to 1.
	25	endrep		n:= n + 1		End loop.
	26	endrep		next m		
postamble	27					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:= 4; 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 channel management command.
	4	await		time:= ctime + (100 + 15) $\times$ 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) $\times$ 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) $\times$ 60/M1 + 60 $\times$ 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) $\times$ 60/M1 + 60 $\times$ 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:= 1; 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 channel management 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.

	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.

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**

	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:= 1; 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 channel management 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					Bring test equipment into idle state.
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	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	Velocity		Input velocity as vel:= 3 073 knots		Input velocity to the station under test that is above 3 069 knots.
	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:= 1; pc:= 0; sc:= 1; TRANS: id(1):= id_val; aux(1):= aux_val; 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 channel management 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_SINGLE_SLOT_TCP_a and b/g = 0, no = 2
				IF (id_val, aux_val, bg_val) = (2hex, 0, 1) THEN SYNC BURST = SYNC_HIGH_DYNAMIC_a and b/g = 1, br/gr = 1
					SHa, SSa	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	Velocity		Input velocity as vel:= 3 069 knots		Input velocity to the station under test of 3 069 knots.
	14	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**
	15	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.

	16	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, 0, 1) 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.

	17	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.
	18	verify	RF	SYNC BURST = SYNC_HIGH_DYNAMIC_a (s = add_A) contains b/g = 0, br/gr = 0	SHa	Verify that b/g and br/gr are set correctly.
	19	endrep		n:= n + 1		End loop.
	20	endrep		next id_val, aux_val, bg_val		Go to next value of (id_val, aux_val, bg_val).
postamble	21					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 parameter, 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:= 4; 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 channel management command.
	4	await		time:= ctime + (100 + 15) $\times$ 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) $\times$ 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) $\times$ 60/M1 + 60 $\times$ 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) $\times$ 60/M1 + 60 $\times$ 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:= 1; 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 channel management 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.
	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 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 (GSC 1)	CTRL_CHAN_I (s:= add_G; HEADER: dt:= 5; do:= 0; dc:= 0; csid:= 1; tc:= 1; fc:= 1; styp:= 0; sr:= 3; pc:= 0; sc:= 2; TRANS: id:= 0; 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)	CI	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 31		n:= 1		Start loop.
	4	await	RF (GSC 1)	SYNC BURST (s = add_A)	Sb	Wait for a sync burst from the station under test.
	5	record	RF (GSC 1)	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(31) - 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 GSC channel (i.e. 6 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 parameter 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 channel management parameter 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:= 4; 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 $\times$ 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 = priority 1.
	8	send	RF	CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 4; 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 $\times$ 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 = priority 2.
	9	await		time:= send_time + 60 + 2 249 $\times$ 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 $\times$ 60/M1		Verify that the station under test transmits the sync burst with full position variable part, and therefore with the higher priority.

Test Case Name:				Channel_Param_Q1		
Purpose:				**To demonstrate that a station in receipt of a channel management parameter will transmit all of the streams defined in the script while applying the Q1 parameter.**		
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	11	await		time:= send_time + 120 + 2 249 × 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 × 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.
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 parameter 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:= 4; 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.

Test Case Name:				Channel_Param_Q4		
Purpose:				**To demonstrate that a station in receipt of a channel management parameter will transmit all of the streams defined in the script while applying the Q4 parameter.**		
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	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:= 4; 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 parameter 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 x 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 x 60/M1		Wait for the first available slot.
	6	send	RF	CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 4; 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.

Test Case Name:	Channel_Param_Q2					
Purpose:	**To demonstrate that a station in receipt of a channel management parameter will transmit all of the streams defined in the script while applying the Q2 parameters.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	14	send	RF	CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 4; 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 parameter 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.

Test Case Name:				Channel_Param_V12		
Purpose:				**To demonstrate that a station in receipt of a channel management parameter will transmit all of the streams defined in the script while applying the V12 parameter.**		
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	8	send	RF	CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 4; 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) x 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.
	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.

Test Case Name:	Channel_Param_V12					
Purpose:	**To demonstrate that a station in receipt of a channel management parameter will transmit all of the streams defined in the script while applying the V12 parameter.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	21	send	RF	CTRL_CHAN_m (s:= add_G; HEADER: dt:= 4; do:= 0; dc:= 4; 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) x 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 parameter 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:= 4; 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 parameter 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:= 4; 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 parameter 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 parameter 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:= 4; 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 parameter 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:= 4; 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 × 0,163LON(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		
	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:= 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.
	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 shall be restarted until this happens.
	12	verify verify verify	RF RF RF	In fixed part of SYNC_BURST_g (s= add_A): cprf = CPR_ENC_TABLE (p, cpr_type) AND lat = CPR_ENC_TABLE (p, lat_enc) AND lon = CPR_ENC_TABLE (p, lon_enc)	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.

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**
	13			In variable part of SYNC_BURST_g (s= add_A):	Sg,	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	pid = CPR_ENC_TABLE (p, pid_enc)	CE(r, c)	
				AND		
		verify	RF	lat6(bits 1-5) = CPR_ENC_TABLE (p, lat6(1-5))		
				AND		
		verify	RF	lat6(bit 6) = CPR_ENC_TABLE (p, lat6(6)		
				AND		
		verify	RF	lon6(bits 1-5) = CPR_ENC_TABLE (p, lon6(1-5))		
				AND		
		verify	RF	lon6(bit 6) = CPR_ENC_TABLE (p, lon6(6))		
	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			In variable part of SYNC_BURST_h (s= add_A):	Sh,	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(bits 1-3) = CPR_ENC_TABLE (p, lat4(1-3))	CE(r, c)	
				AND		
		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:= 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.
	18	await	RF	SYNC_BURST_i (s= add_A)	Si	
	19			In variable part of SYNC_BURST_i (s= add_A):	Si,	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(bits 1 to 7) = CPR_ENC_TABLE (p, lat8(1-7))	CE(r, c)	
				AND		
		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))		
	20	endrep		n:= n + 1		
postamble	21	send	VSS	SET PARAMETERS (p:= 64/256)		Restore to default value.
	22	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_j (po:= 0; pt:= 0; s:= add_B;
				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)))
					Sj, CE (r, c)	Send a sync burst from a simulated station B with a high dynamic 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 high resolution 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.
	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 aground 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 *RFIN*PARAMETERS(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 *RFIN*PARAMETERS(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 ADS-B 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 [i.1];
- column 3 identifies individual requirements within ICAO VDL4 Technical Manual [i.1];
- column 4 identifies clause titles taken from the present document;
- column 5 is a reference to testing requirements specified elsewhere in the present document. Several tests verify a whole group of requirements. They are only mentioned in the first row of such a group, usually a headline. The applicability of these tests to the subordinated requirements is indicated by ditto marks (") in the rows following the first instance of a test case name. Amplification of individual entries is provided by the following notes;
- column 6 qualifies each test procedure to be:
  - Essential: meaning that it is included with the Essential Radio Test Suite and therefore the requirement should be demonstrated to be met in accordance with the referenced procedures.
  - Other: meaning that the test procedure is illustrative but other means of demonstrating compliance with the requirement are permitted.
  - eXcluded: meaning that there is no specific test for the requirement.

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

Requirement reference	Reference in [i.1]	Req	Title	Test Case	E/O/X
5.1.1	3.3		Information Field Formats	See note 1	X
5.1.1.1	3.3	a		See note 1a	X
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	E
5.1.1.3	3.3.3	a		See note 1a	X
5.1.1.4	3.3.1	a		See note 1a	X
5.1.1.5	3.3.3.2	a		See note 1a	X
5.1.1.5a				See note 2	O
5.1.1.6	3.3.3.2	b		See note 1a	X
5.1.1.7	3.3.4	a		See note 1a	X

Requirement reference	Reference in [i.1]	Req	Title	Test Case	E/O/X
5.1.1.8	3.3.5	a		See note 1a	X
5.1.1.9	3.3.6	a		See note 1a	X
5.1.1.10	3.3.7	a		See note 1a	X
5.1.1.11	3.3.8	a		See note 1a	X
5.1.1.12	3.3.9	a		See note 1a	X
5.1.1.13				See note 1a	X
5.1.1.14				See note 1a	X
5.1.2	3.4		ADS-B Request	See note 1	X
5.1.2.1	3.4.1		ADS-B Request Format	See note 1	X
5.1.2.1.1	3.4.1	a		ADSB_Request_A	E
5.1.2.1.2	3.4.1	b		ADSB_Request_A ADSB_Request_B ADSB_Request_C ADSB_Request_D	E
5.1.2.1.3	3.4.1	c		See note 1a	X
5.1.2.1.4	3.4.1	d		See note 1a	X
5.1.2.1.5	3.4.1	e		See note 1a	X
5.1.2.2	3.4.2		Sleep Mode	See note 1	X
5.1.2.2.1	3.4.2	a		ADSB_Request_C ADSB_Request_D	E
5.1.2.2.2	3.4.2	b		ADSB_Request_A ADSB_Request_B	E
5.1.2.3	3.4.3		Automatic Selection of Variable Information Fields	See note 1	X
5.1.2.3.1	3.4.3	a		ADSB_Request_A ADSB_Request_C	E
5.1.2.3.2	3.4.3	b		ADSB_Request_A ADSB_Request_C	E
5.1.2.3.3	3.4.3	c		ADSB_Request_B ADSB_Request_D	E
5.1.2.3.4	3.4.3	d		ADSB_Request_B ADSB_Request_D	E
5.1.3	3.5		Default ADS-B Reporting	See note 1	X
5.1.3.1	3.5	a		ADSB_reporting	E
5.1.3.1a				See note 2	O
5.1.3.1b				See note 2	O
5.1.3.2	3.5.1	a		See note 4	O
5.1.4	3.6		ADS-B Procedures	See note 1	X
5.1.4.1	3.6.1.1	a		ADSB_Request_B ADSB_Request_D	E
5.1.4.2	3.6.1.1	b		ADSB_Request_A ADSB_Request_C	E
5.1.4.3	3.6.1.1	c		ADSB_Request_A ADSB_Request_B	E
5.1.4.4	3.6.1.2	a		ADSB_Request_C ADSB_Request_D	E
5.1.4.4a	3.6.1.2	a		See note 2	O
5.1.4.4b	3.6.1.3	a		ADSB_Sleep_B	E
5.1.4.4c	3.6.1.3	b		ADSB_Sleep_B	E
5.1.4.4d	3.6.1.3	c		ADSB_Sleep_A	E
5.1.4.5	3.6.2	a		ADSB_request_time	E
5.1.4.6	3.6.3	a		ADSB_priority	E
5.1.4.6a	3.6.4.1	a		ADSB_First_TCP_A	E
5.1.4.6b	3.6.4.1	b		ADSB_First_TCP_A	E
5.1.4.6c	3.6.4.1	c		ADSB_First_TCP_A	E
5.1.4.6d	3.6.4.2	a		ADSB_Status_SVQ_A	E
5.1.4.6e	3.6.4.2	b		ADSB_Status_SVQ_A	E
5.1.4.6f	3.6.4.3	a		ADSB_TCP/SVQ_Flag_A	E
5.1.4.7	3.6.4.4	a		ADSB_Status_SVQ_A	E
5.1.4.8	3.6.4.5	a		ADSB_TCP/SVQ_Flag_A	E
5.1.4.9	3.6.4.6	a		ADSB_TCP/SVQ_Flag_B	E
5.1.4.10				See note 2	O
5.1.4.11				See note 2	O
5.1.5	3.7		CTRL Parameters	See note 1	X

Requirement reference	Reference in [i.1]	Req	Title	Test Case	E/O/X
5.1.5.1	3.7.1		Directory of Service (DOS) Message	See note 1	X
5.1.5.1.1	3.7.1.1	a		See note 3	X
5.1.5.1.2	3.7.1.1	b		See note 1a	X
5.1.5.1.3	3.7.1.2	a		See note 1a	X
5.1.5.2	3.7.2		Channel Management Parameter	See note 1	X
5.1.5.2.1	1.5.3.6.7	a		See note 1a	X
5.1.5.2.2	1.5.3.6.7	b		See note 1a	X
5.1.5.2.3	3.7.2	a		See note 1a	X
5.1.5.2.4	3.7.2	b		See note 1a	X
5.1.5.2.5	3.7.2.1.1	a		See note 1a	X
5.1.5.2.6	3.7.2.1.2	a		See note 1a	X
5.1.5.2.7	3.7.2.1.2	b		See note 1a	X
5.1.5.2.8	3.7.2.2	a		See note 1a	X
5.1.5.2.9	3.7.2.2.1	a		Channel_Dest_Count	E
5.1.5.2.10	3.7.2.2.2	a		Channel_Dir_Rate_A Channel_Dir_Rate_B Channel_Directed	E
5.1.5.2.10a	3.7.2.2.2	b		Channel_Address	E
5.1.5.2.11	3.7.2.2.3.1	a		Channel_Dir_Rate_A Channel_Dir_Rate_B	E
5.1.5.2.12	3.7.2.2.3.2	a		Channel_Dest_Count	E
5.1.5.2.13	3.7.2.2..4.1	a		Channel_Directed	E
5.1.5.2.14	3.7.2.2..4.1	b		Channel_Directed	E
5.1.5.2.16	3.7.2.2.5	a		Channel_Multicast_A Channel_Multicast_B Channel_Multicast_C Channel_Multicast_D	E
5.1.5.2.16a	3.7.2.2.5	b		Channel_Multicast_A	E
5.1.5.2.17	3.7.2.2.5	c		Channel_Multicast_B Channel_Multicast_C Channel_Multicast_D	E
5.1.5.2.17a	3.7.2.2.5	d		Channel_Multicast_B Channel_Multicast_C Channel_Multicast_D	E
5.1.5.2.17b	3.7.2.3.1.1	a		Channel_Dir_Rate_A Channel_Dir_Rate_B Channel_Directed	E
5.1.5.2.17c	3.7.2.3.1.1	b		Channel_Dir_Rate_A Channel_Directed	E
5.1.5.2.17d	3.7.2.3.1.1	c		Channel_Dir_Rate_B	E
5.1.5.2.17e	3.7.2.3.1.1	d		Channel_Invalid	E
5.1.5.2.17f	3.7.2.3.1.1	e		Channel_Invalid	E
5.1.5.2.17g	3.7.2.3.1.2	a		Channel_Dir_Rate_A Channel_Dir_Rate_B Channel_Directed	E
5.1.5.2.17h	3.7.2.3.1.2	b		See note 1a	X
5.1.5.2.17i	3.7.2.3.2	a		Channel_Dir_Rate_A Channel_Dir_Rate_B	E
5.1.5.2.17j	3.7.2.3.2	b		Channel_Dir_Rate_A Channel_Dir_Rate_B	E
5.1.5.2.17k	3.7.2.3.2	c		Channel_Dir_Rate_A Channel_Dir_Rate_B	E
5.1.5.2.17l	3.7.2.3.3	a		Channel_Directed	E
5.1.5.2.17m	3.7.2.3.3	b		Channel_Directed	E
5.1.5.2.17n	3.7.2.3.3	c		Channel_Directed	E
5.1.5.2.17o	3.7.2.3.4	a		Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B	E
5.1.5.2.17p	3.7.2.3.4	b		Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B	E
5.1.5.2.17q	3.7.2.3.4	c		Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B	E

Requirement reference	Reference in [i.1]	Req	Title	Test Case	E/O/X
5.1.5.2.17r	3.7.2.3.4	d		Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B	E
5.1.5.2.17s	3.7.2.3.4	e		Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B	E
5.1.5.2.17t	3.7.2.3.4	f		Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B	E
5.1.5.2.17u	3.7.2.3.5.1	a		Channel_Dir_Rate_B	E
5.1.5.2.17v	3.7.2.3.5.1	b		Channel_Frequency	E
5.1.5.2.17w	3.7.2.3.5.1	c		Channel_Frequency	E
5.1.5.2.17x	3.7.2.3.5.1	d		Channel_Length	E
5.1.5.2.17y	3.7.2.3.5.2	e		Channel_Plea	E
5.1.5.2.18	3.7.2.4.1	a		Channel_Cancel_DT	E
5.1.5.2.19a	3.7.2.4	d		Channel_Cancel_DT Channel_Cancel_CSID_A Channel_Cancel_CSID_B	E
5.1.5.2.19b	3.7.2.5	a		Channel_Cancel_Auto	E
5.1.5.2.19c	3.7.2.5.1.1	a		Channel_Cancel_CSID_C	E
5.1.5.2.19d	3.7.2.5.1.2	a		Channel_Cancel_CSID_D	E
5.1.5.2.19e	3.7.2.5.1.3	a		Channel_Directed Channel_Dir_Rate_A Channel_Dir_Rate_B	E
5.1.5.2.19f	3.7.2.5.1.4	a		Channel_Cancel_CSID_A Channel_Cancel_CSID_B	E
5.1.5.2.20	3.7.2.6.1	a		Channel_Directed Channel_Dir_Rate_A	E
5.1.5.2.20a	3.7.2.6.1	b		Channel_Unrecognize	E
5.1.5.2.21	3.7.2.6.2	a		Channel_User_Trans Channel_Non_Sync_Trans	E
5.1.5.2.22	3.7.2.6.2	b		Channel_User_Trans Channel_Non_Sync_Trans	E
5.1.5.2.22a	3.7.2.6.2.2	a		Channel_Speed	E
5.1.5.2.22b	3.7.2.6.2.3	a		Channel_NotSupported	E
5.1.5.2.22c	3.7.2.7	a		Channel_Frequency	E
5.1.5.2.22d	3.7.2.7	b		See note 2	O
5.1.5.2.23	3.7.2.7	c		Channel_Frequency	E
5.1.5.2.23a	3.7.2.7.1	a		See note 4	O
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	W
5.1.5.2.24a	3.7.2.8	b		See note 2	O
5.1.5.2.24b	3.7.2.8	c		See note 2	O
5.1.5.2.24c	3.7.2.8	d		See note 2	O
5.1.5.2.24d				See note 2	O
5.1.5.2.25	3.7.2.9.1	a		Channel_Dir_Rate_A Channel_Dir_Rate_B Channel_User_Trans	E
5.1.5.2.26	3.7.2.9.2	b		Channel_Dir_Rate_A Channel_Dir_Rate_B Channel_User_Trans	E
5.1.5.2.27	3.7.2.9.2	c		Channel_Script_Repeat	E
5.1.5.2.28	3.7.2.9.2	d		See note 2	O
5.1.5.2.29	3.7.2.9.2	e		Channel_Discard	E
5.1.6	4		Definitions for Compact Position Reporting	See note 1	X
5.1.6.1	4.6		Information Field Offset Encoding	See note 1	X
5.1.6.1.1	4.6.1	a		CPR_Encode_Offset	E
5.1.6.1.2	4.6.1	b		CPR_Encode_Offset	E
5.1.6.1.3	4.6.2	a		See note 1a	X
5.1.6.1.4	4.6.3.1	a		CPR_Encode_Offset	E

Requirement reference	Reference in [i.1]	Req	Title	Test Case	E/O/X
5.1.6.1.5	4.6.3.2	a		CPR_Encode_Offset	E
5.1.6.1.6	4.6.3.3	a		CPR_Encode_Offset	E
5.1.6.2	4.7		Information Field Offset Decoding	See note 1	X
5.1.6.2.1	4.7.1	a		CPR_Decode_Offset	E
5.1.6.2.2	4.7.2	a		See note 1a	X
5.1.6.2.3	4.7.3.1	a		CPR_Decode_Offset	E
5.1.6.2.4	4.7.3.2	a		CPR_Decode_Offset	E
5.1.6.3	4.8		Patch ID Encoding	See note 1	X
5.1.6.3.1	4.8.1	a		CPR_Encode_Offset	E
5.1.6.3.2	4.8.2	a		See note 1a	X
5.1.6.3.3	4.8.3	a		See note 1a	X
5.1.6.4	4.9		Patch ID Decoding	See note 1	X
5.1.6.4.1	4.9.1	a		CPR_Decode_Offset	E
5.1.6.4.2	4.9.2	a		See note 1a	X
5.1.6.4.3	4.9.3.1	a		CPR_Decode_Offset	E
5.1.6.4.4	4.9.3.2	a		CPR_Decode_Offset	E
5.1.7			Encoding of UDATA (udid)	See note 1	X
5.1.7.1				See note 1a	X

Table A.2 outlines the mapping between the VDL Mode 4 airborne station TIS-B requirements and the related test procedures. In these tables:

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

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

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

Table A.3 outlines the mapping between the VDL Mode 4 airborne station FIS-B requirements and the related test procedures. In these tables:

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

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

Requirement reference	Title	Test Case	E/O/X
5.3	Requirements for FIS-B	See note 1	X
5.3.1	Message format	See note 1	X
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	E
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	E
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	E
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	E
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	E

Requirement reference	Title	Test Case	E/O/X
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	E
5.3.2	Meteorological Aerodrome Report (METAR) message	See note 1	X
5.3.2.1		FIS_METAR_A	E
5.3.2.2		FIS_METAR_C	E
5.3.2.3		FIS_METAR_A	E
5.3.2.4		FIS_METAR_A	E
5.3.2.5		FIS_METAR_A	E
5.3.2.6		FIS_METAR_A	E
5.3.2.7		FIS_METAR_A	E
5.3.2.8		FIS_METAR_A	E
5.3.2.9		FIS_METAR_A	E
5.3.3	Special Observations and Reports (SPECI) message	See note 1	X
5.3.3.1		FIS_SPECI_A	E
5.3.4	Automatic Terminal Information Service (ATIS) message	See note 1	X
5.3.4.1		FIS_ATIS_A	E
5.3.4.2		FIS_ATIS_A	E
5.3.4.3		FIS_ATIS_A	E
5.3.4.4		FIS_ATIS_A	E
5.3.4.5		FIS_ATIS_A	E
5.3.4.6		FIS_ATIS_A	E
5.3.4.7		FIS_ATIS_A	E
5.3.5	Runway Condition (RCN) message	See note 1	X
5.3.5.1		FIS_RCN_A	E
5.3.5.2		FIS_RCN_A	E
5.3.6	SIGMET Reports	See note 1	X
5.3.6.1		FIS_SIGMET_A	E
5.3.6.2		FIS_SIGMET_A	E
5.3.6.3		FIS_SIGMET_A	E
5.3.6.4		FIS_SIGMET_A	E
5.3.7	Temporary Segregated Areas (TSA) message	See note 1	X
5.3.7.1		FIS_TSA_CTR_A FIS_TSA_DNG_A	E
5.3.7.2		FIS_TSA_CTR_A FIS_TSA_DNG_A	E
5.3.7.3		FIS_TSA_CTR_A	E
5.3.7.4		FIS_TSA_DNG_A	E
5.3.7.5		FIS_TSA_CTR_A FIS_TSA_DNG_A	E
5.3.8	FIS report request message	See note 1	X
5.3.8.1		FIS_Request_Rec_A	E
5.3.8.2		FIS_Request_Rec_A	E
5.3.8.3		FIS_Request_Rec_A	E
5.3.8.4		FIS_Request_Rec_A	E
5.3.8.5		FIS_Request_Rec_A	E
5.3.8.6		FIS_Request_Rec_A	E
5.3.8.7		FIS_Request_Rec_A	E
5.3.9a	Airborne system functions	See note 1	X
5.3.9a.2	Transmission of requests	See note 1	X
5.3.9a.2.1		See note 2	O
5.3.9a.3	Differentiation between reports	See note 1	X
5.3.9a.3.1		See note 2	O

Table A.4 outlines the mapping between the VDL Mode 4 airborne station GNS-B requirements and the related test procedures. In these tables:

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

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

Requirement reference	Title	Test Case	E/O/X
5.4	Requirements for GNSS Augmentation Service Broadcast (GNS-B) message	See note 1	X
5.4.1	Message format	See note 1	X
5.4.1.1		GNS_Type1_Message_A GNS_Type2_Message_A GNS_Type4_Message_A	E
5.4.1.2		GNS_Type1_Message_A GNS_Type2_Message_A GNS_Type4_Message_A	E
5.4.1.3		GNS_Type1_Message_A GNS_Type2_Message_A GNS_Type4_Message_A	E
5.4.1.4		GNS_Type1_Message_A GNS_Type2_Message_A GNS_Type4_Message_A	E
5.4.1.5		GNS_Type1_Message_A GNS_Type2_Message_A GNS_Type4_Message_A	E
5.4.2	Message Type 1	See note 1	X
5.4.2.1		GNS_Type1_Message_A	E
5.4.2.2		GNS_Type1_Message_A	E
5.4.2.3		GNS_Type1_Message_A	E
5.4.2.4		See note 2	O
5.4.2.5		See note 1a	X
5.4.2.6		See note 2	O
5.4.3	Message Type 2	See note 1	X
5.4.3.1		GNS_Type2_Message_A	E
5.4.3.2		GNS_Type2_Message_A	E
5.4.4	Message Type 4	See note 1	X
5.4.4.1		GNS_Type4_Message_A	E
5.4.4.2		GNS_Type4_Message_A	E
5.4.4.3		GNS_Type4_Message_A	E
5.4.4.3a		See note 1a	X
5.4.4.4		See note 2	O
5.4.4.5		See note 2	O
5.4.5	CRC Calculation	See note 1	X
5.4.5.1	GNS-B CRC	See note 1	X
5.4.5.1.1		See note 1a	X
5.4.5.1.2		See note 1a	X

Requirement reference	Title	Test Case	E/O/X
5.4.5.1.3		See note 1a	X
5.4.5.1.4		See note 1a	X
5.4.5.1.5		See note 1a	X
5.4.5.2	FAS CRC	See note 1	X
5.4.5.2.1		See note 1a	X
5.4.5.2.2		See note 1a	X
5.4.5.2.3		See note 1a	X
5.4.5.2.4		See note 1a	X
5.4.5.2.5		See note 1a	X
5.4.5.3	Ephemeris CRC	See note 1	X
5.4.5.3.1		See note 1a	X
5.4.5.3.2		See note 1a	X
5.4.5.3.3		See note 1a	X
5.4.5.3.4		See note 1a	X
5.4.5.3.5		See note 1a	X
5.4.5.3.6		See note 1a	X
5.4.5.3.7		See note 1a	X
5.4.5.3.8		See note 1a	X
5.4.5.3.9		See note 1a	X

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

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

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

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

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

Annex B (informative): Bibliography

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.net/.

History

Document history		
V1.1.1	July 2005	Publication
V1.2.1	December 2006	Publication
V1.3.1	September 2011	Publication
V1.4.0	December 2014	EN Approval Procedure AP 20150404: 2014-12-05 to 2015-04-06
V1.4.1	April 2015	Publication