Final draft ETSI EN 302 842-2 V1.2.1 (2006-07)

European Standard (Telecommunications series)

Electromagnetic compatibility and Radio spectrum Matters (ERM); VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment; Part 2: General description and data link layer

Reference
REN/ERM-TG25-030-2

Keywords
aeronautical, digital, radio, testing, VHF

ETSI

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

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

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

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

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

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

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Contents

Intellectual Property Rights

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

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

Foreword

This European Standard (Telecommunications series) has been produced by ETSI Technical Committee Electromagnetic compatibility and Radio spectrum Matters (ERM), and is now submitted for the ETSI standards One-step Approval Procedure.

The present document is part 2 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";

Part 5: "Description of compliance with the SES 552/2004 interoperability Regulation".

The present document is accompanied by an equivalent ground-based standard, EN 301 842 [15] parts 1 to 6, covering the VHF air-ground Data Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for ground-based equipment.

NOTE: Minimum Operational Performance Specifications (MOPS) are also being developed for VDL Mode 4. EUROCAE have previously published Interim MOPS for VDL Mode 4 (see bibliography) which are a sub set of EN 302 842-1, 2, 3 and 4. EN 302 842-1, 2, 3 and 4 complies with the requirements of CEC Mandate M/318.

Proposed national transposition dates	
Date of latest announcement of this EN (doa):	3 months after ETSI publication
Date of latest publication of new National Standard or endorsement of this EN (dop/e):	6 months after doa
Date of withdrawal of any conflicting National Standard (dow):	6 months after doa

Introduction

The present document states the technical specifications for Very High Frequency (VHF) Digital Link (VDL) Mode 4 aeronautical mobile (airborne) radio transmitters, transceivers and receivers for air-ground and air-air communications operating in the VHF band, using Gaussian Filtered Frequency Shift Keying (GFSK) modulation with 25 kHz channel spacing and capable of tuning to any of the 25 kHz channels from 118,000 MHz to 136,975 MHz as defined in ICAO VDL SARPs [2].

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

The present document has been written on the assumption that:

- the type test measurements will be performed only once, in an accredited test laboratory and the measurements accepted by the various authorities in order to grant type approval;
- if equipment available on the market is required to be checked it will be tested in accordance with the methods of measurement specified in the present document or a documented alternative approved by the certifying authority;
- equipment comply with EN 302 842-1 [4].

The present document also indicates VDL Mode 4 compliance with the SES 552/2004 interoperability Regulation [14].

NOTE: The present document has been produced with a view to maintaining consistency of numbering with the equivalent standard for ground equipment (see EN 301 842 [15]). Where requirements are the same, they have been given the same number. Some new airborne requirements have been inserted between requirements that were sequential in EN 301 842 [15]. 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 118,000 MHz to 136,975 MHz as defined in ICAO VDL SARPs [2].

The present document provides part 2 of the technical specifications.

The present document is designed to ensure that equipment certified to it will be compatible with the relevant ICAO VDL SARPs [2] and ICAO VDL4 Technical Manual [1] and with the SES 552/2004 interoperability Regulation.

NOTE: In clause 5.1.4, Tertiary time, as described by the ICAO VDL4 Technical Manual [1], has not been included as a requirement in the present document due to the opinion of the ETSI Working Group that insufficient evidence was available to be able to verify correct operation of a "floating network" of stations operating on tertiary time.

Manufacturers should note that in future the tuning range for the transmitter may also cover any 25 kHz channel from 112,000 MHz to 117,975 MHz and the receiver(s) may cover any 25 kHz channel from 108,000 MHz to 117,975 MHz.

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

The scope of the present document is limited to mobile stations. The equivalent specification for ground stations is EN 301 842 [15].

The VDL Mode 4 system provides digital communication exchanges between aircraft and ground based systems and other aircraft supporting surveillance and communication applications. The supported modes of communication include:

- broadcast and point-to-point communication;
- broadcast services including Automatic Dependent Surveillance - Broadcast (ADS-B), Traffic Information Service - Broadcast (TIS-B), Flight Information Service - Broadcast (FIS-B) capabilities and GNSS Augmentation Service - Broadcast (GNS-B);
- air-air and ground-air services;
- operation without ground infrastructure.

VDL Mode 4 is designed to be an Air/Ground subsystem of the Aeronautical Telecommunication Network (ATN) (see ICAO ATN SARPs [8]) using the AM(R)S band and it is organized according to the Open Systems Interconnection (OSI) model (defined by ISO). It provides reliable sub network services to the ATN system. Other networks can also be supported but have not been the focus of the present document.

The present document is derived from the standards and specifications in:

- ICAO VDL4 Technical Manual [1] produced under the auspices of the International Civil Aviation Organization (ICAO).
- Other relevant standards as defined in clause 2.

It is envisaged that manufacturers may provide equipment supporting:

- broadcast services only;
- point-to-point services only;
- both broadcast and point-to-point services.

EN 302 842-1 [4] deals with tests of the physical layer. The present document defines the core link layer requirements for the VDL Mode 4 mobile station necessary to support all types of equipment. This includes a simple position broadcast functionality.

The present document deals with tests of the link layer sufficient to support core link layer functionality, and it also includes requirements and tests sufficient to recognize and respond to transmissions associated with point-to-point communication. The present document does not address requirements for the full ADS-B message set, or for other broadcast applications that can be supported by the VDL Mode 4 equipment. These are covered by EN 302 842-3 [9]. Detailed requirements for point-to-point communication are beyond the scope of the present document, but can be found in EN 302 842-4 [10]. EN 302 842-4 [10] also includes the interface to the Aeronautical Telecommunication Network (ATN) as defined in ICAO ATN SARPs [8].

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 link layer;
- Clause 5 provides performance specifications for the VDL Mode 4 mobile station and any additional mobile functions necessary to support ground station co-ordination;
- Clause 6 provides general design requirements;
- Clause 7 provides protocol tests for core link layer functions;
- Annex A provides a detailed cross-reference to the relevant requirements contained in ICAO VDL4 Technical Manual [1];
- Annex B provides a description of the ISO/IEC 9646 [7] test methodology;
- Annex C provides a Bibliography;
- a document history.

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

- a detailed set of protocol tests are provided for the core link layer functionality necessary to support broadcast functions;
- a detailed test of position encoding and decoding is provided because of the importance of position in the management of the VDL Mode 4 link specifically and the need to support ADS-B applications in general.

Mandating and Recommendation Phrases

a) "Shall":

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

b) "Should":

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

2 References

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

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

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

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

[2] ICAO Annex 10 to the Convention on International Civil Aviation: "Aeronautical Telecommunications, Volume III: Communication Systems, Part I: Digital Data Communication Systems, Chapter 6".

[3] ISO/IEC 13239 (2002): "Information technology - Telecommunications and information exchange between systems - High-level data link control (HDLC) procedures".

[4] ETSI EN 302 842-1 (V1.2.1): "Electromagnetic compatibility and Radio spectrum Matters (ERM); VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment; Part 1: Physical Layer".

[5] ISO/IEC 7498-1 (1994): "Information technology - Open Systems Interconnection - Basic Reference Model: The Basic Model".

[6] ISO/IEC 10731 (1994): "Information technology - Open Systems Interconnection - Basic Reference Model - Conventions for the definition of OSI services".

[7] ISO/IEC 9646 (all parts): "Information technology - Open Systems Interconnection - Conformance testing methodology and framework".

[8] ICAO 9705-CD: "Manual of Technical Provisions for the Aeronautical Telecommunication Network (ATN)".

[9] ETSI EN 302 842-3 (V1.2.1): "Electromagnetic compatibility and Radio spectrum Matters (ERM); VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment; Part 3: VDL Mode 4 airborne equipment additional broadcast aspects".

[10] ETSI EN 302 842-4 (V1.2.1): "Electromagnetic compatibility and Radio spectrum Matters (ERM); VHF air-ground and air-air Digital Link (VDL) Mode 4 radio equipment; Technical characteristics and methods of measurement for aeronautical mobile (airborne) equipment; Part 4: VDL Mode 4 airborne equipment point to point functions".

[11] EUROCAE ED-14D / RTCA DO-160D: "Environmental Conditions and Test Procedures for Airborne Equipment, July 1997, as amended by Change 1 (December 2000), by Change 2 (June 2001), and by Change 3 (December 2002)".

[12] Eurocontrol (2003): "VDL Mode 4 Airborne Architecture Study documentation".

[13] EUROCAE ED-12B / RTCA DO-178B (1993): "Software Considerations in Airborne Systems and Equipment Certification".

[14] Regulation (EC) No 552/2004 of the European Parliament and of the Council of 10 March 2004 on the interoperability of the European Air Traffic Management network (the interoperability Regulation).

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

3 Definitions and abbreviations

3.1 Definitions

3.1.1 Basic reference model definitions

The present document is based on the concepts developed in the open systems interconnect basic reference model. For the purposes of the present document, the terms and definitions given in ISO/IEC 7498-1 [5] apply:

- layer;
- sublayer;
- entity;
- service;
- physical layer;
- data link layer.

3.1.2 Service conventions definitions

For the purposes of the present document, the terms and definitions given in ISO/IEC 10731 [6] applies for:

- service provider;
- request;
- indication;
- confirm.

3.1.3 General definitions

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

Aeronautical Mobile Service (AMS): mobile service between aeronautical stations and aircraft stations, or between aircraft stations, in which survival craft stations may participate

Aeronautical Telecommunications Network (ATN): internetwork architecture that allows ground, air/ground, and aircraft data sub networks to interoperate by adopting common interface services and protocols based on the International Organization for Standardization Open Systems Interconnection Reference Model

aircraft address: 24-bit address available for assignment to an aircraft for the purpose of air-ground communications, navigation and surveillance

NOTE: An aircraft may choose not to use this unique address and can use instead a non-unique address.

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 situation awareness functionality, enhanced traffic management in the air and on the ground, search and rescue support and others.

autotune: procedure by which a VDL Mode 4 ground station may direct a mobile VDL Mode 4 station to transmit on a specified frequency, and with certain characteristics, by sending an uplink burst containing an autotune reservation

Big Negative Dither (BND): reservation protocol by which a station may establish a reservation in the following superframe outside the range of the maximum dither allowed by periodic broadcasts

NOTE: It is intended to be used by a station entering the network after listening to the channel for a few seconds, and allows such a station to reserve a slot which is likely to be unreserved.

burst length: number of slots across which the VDL Mode 4 burst is transmitted

CTRL DLPDU: basic unit of transmission at the LME and VME

current slot: slot in which a received transmission begins

Data Link Entity (DLE): protocol state machine capable of setting up and managing a single data link connection

Data Link Protocol Data Unit (DLPDU): general burst format used by the Data Link Service (DLS) sublayer

Data Link Service (DLS) Sublayer: sublayer that resides above the VDL Mode 4 Specific Services (VSS) and the MAC sublayers

NOTE: The Data Link Service (DLS) manages the transmit queue, creates and destroys Data Link Entities (DLEs) for connection-oriented communications provides facilities for the Link Management Entity (LME) to manage the DLS, and provides facilities for connection-less communications.

delayed burst: VDL Mode 4 burst that begins sufficiently after the beginning of a slot so that the transmitting VDL Mode 4 station is confident that no other VDL Mode 4 station that it could receive from and is within the guard range is transmitting in the slot. The delayed VDL Mode 4 burst terminates by the end of the slot in which it began (its length is shortened to ensure completion by the nominal time)

Global Signalling Channel (GSC): channel available on a world-wide basis for VDL Mode 4 based services

ground base station: aeronautical station equipment, in the aeronautical mobile service, for use with an external antenna and intended for use at a fixed location

link: link connects a mobile DLE and a ground DLE and is uniquely specified by the combination of mobile DLS address and the ground DLS address

NOTE: A different sub network entity resides above every link endpoint.

link establishment: process by which two stations discover each other, determine to communicate with each other, decide upon the communication parameters, create a link and initialize its state before beginning communications

NOTE: For air-ground links, this process involves ground LME protocols. For air-air links, link establishment is achieved using mobile ZOCOP protocols.

link handoff: process by which peer LMEs, already in communication with each other, create a link between an aircraft and a new ground station before disconnecting the old link between the aircraft and the current ground station

link layer: layer that lies immediately above the physical layer in the Open Systems Interconnection protocol model

NOTE: The link layer provides for the reliable transfer of information across the physical media. It is subdivided into the data link sublayer and the media access control sublayer.

Link Management Entity (LME): protocol state machine capable of acquiring, establishing, and maintaining a connection to a single peer system. In addition, in VDL Mode 4, the LME generates synchronization bursts for transmission, either autonomously, or in accordance with a request from a peer station and/or local application.

NOTE: A LME establishes data link and sub network connections, "hands-off" those connections, and manages the media access control sublayer and physical layer. An aircraft LME tracks how well it can communicate with the ground stations of a single ground system. An aircraft VDL Management Entity (VME) instantiates an LME for each ground station that it monitors. Similarly, the ground VME instantiates an LME for each aircraft that it monitors. An LME is deleted when communication with the peer system is no longer viable.

M2 filter parameter: controls the need or otherwise to execute a network entry procedure as a result of a temporary loss of receiving function

Media Access Control (MAC): sublayer that acquires the data path and controls the movement of bits over the data path

physical layer: lowest level layer in the Open Systems Interconnection protocol model

NOTE: The physical layer is concerned with only the transmission of binary information over the physical medium (e.g. VHF radio).

primary time source: source of timing information local to a mobile station, capable of maintaining synchronization to Universal Coordinated Time (UTC) seconds within a prescribed tolerance

private parameters: parameters that are contained in CTRL and UCTRL DLPDUs and that are unique to the VHF digital link environment

reference bit sequence: sequence of bits used in the transmitter performance specifications

reference signal level: reference signal level is the signal level used in the receiver performance specifications unless otherwise stated

secondary navigation: a means by which a station that loses its source of position information may continue to derive its own position from the time-of-arrival of synchronization bursts received from other stations advertising certified data quality

NOTE: The station can deduce the distance of another station from the time-of-arrival of its sync burst, and it knows the position of that station from the positioning information contained in the sync burst

secondary time source: timing source used in a failure mode, that applies when the primary time source fails, in which a VDL Mode 4 station maintains time synchronization to the UTC second

Self-Organizing Time Division Multiple Access (STDMA): multiple access scheme based on time-shared use of a radio frequency (RF) channel employing: (1) discrete contiguous time slots as the fundamental shared resource; and (2) a set of operating protocols that allows users to access these time slots in an organized manner without reliance on a master control station

slot: one of a series of consecutive time intervals of equal duration. Each burst transmission starts at the beginning of a slot (with the exception of VDL Mode 4 delayed transmissions). In VDL Mode 4, each group of slots of one second duration is aligned to the UTC second.

station: VDL Mode 4 Specific Services (VSS)-capable entity

NOTE: A station may be either a mobile station or a ground station. A station is a physical entity that transmits and receives bursts over the RF interface (either A/G or air-to-air (A/A)) and comprises, at a minimum: a physical layer, media access control sublayer, and a unique VSS address. A station which is also a DLS station has the same address.

subnetwork layer: layer that establishes, manages, and terminates connections across a subnetwork

superframe: group of slots that span a period of one UTC minute

NOTE: The start of the current superframe is aligned with the UTC-minute.

synchronization burst (or "sync" burst): VDL Mode 4 burst type containing, as a minimum, information on the station's identity, position and time. A synchronization burst may also carry additional data elements required for specific applications

NOTE: Ground stations announce existence, position, and the current time. Mobile stations lacking timing information can then derive the slot structure and time from ground synchronization bursts. Mobile stations lacking position information can derive position from both mobile and ground synchronization bursts. This periodic information is used in various ways including ADS-B, secondary navigation, and simplifying the LME algorithms.

Time Division Multiple Access (TDMA): multiple access scheme based on time-shared use of an RF channel employing: (1) discrete contiguous time slots as the fundamental shared resource; and (2) a set of operating protocols that allows users to interact with a master control station to mediate access to the channel

VDL Management Entity (VME): VDL-specific entity that provides the quality of service requested by the ATN-defined sub network system management entity

NOTE: A VME uses the LMEs (that it creates and destroys) to acquire the quality of service available from peer systems.

VDL Mode 4: data link using a Gaussian Filtered Frequency Shift Keying modulation scheme and self-organizing time division multiple access

VDL Mode 4 burst: sequence of source address, burst ID, information, slot reservation, and Cyclic Redundancy Check (CRC) fields, bracketed by opening and closing flag sequences

NOTE: The start of a burst may occur only at the start of a VDL Mode 4 time slot, and this constraint allows the propagation delay between the transmission and reception to be derived.

VDL Mode 4 Specific Services (VSS) sublayer: sublayer that resides above the MAC sublayer and provides VDL Mode 4 specific access protocols including reserved, random and fixed protocols

VDL Mode 4 station: physical entity that transmits and receives VDL Mode 4 bursts over the RF interface (either air-ground (A/G), air-to-air (A/A) or ground-ground (G/G))

NOTE 1: It comprises, as a minimum: a physical layer, Media Access Control sublayer and a VSS sublayer.

NOTE 2: A VDL Mode 4 station may either be a mobile VDL Mode 4 station or a ground VDL Mode 4 station.

VDL Mode 4 station address: 27-bit identifier used to identify a VDL Mode 4 station, which may be unique or locally unique. A combination of the 24 bit ICAO aircraft address plus 3 additional bits to discriminate between unique/non-unique and ground/mobile addresses

VDL station: VDL-capable entity.

NOTE: A station may either be a mobile station or a ground station. A station is a physical entity that transmits and receives messages over the air-air and air-ground interface and comprises, at a minimum: a physical layer, a media access control sublayer, and a unique station address.

VDL system: VDL-capable entity

NOTE: A system comprises one or more stations and the associated VDL management entity. A system may either be a mobile system or a ground system.

VSS user: user of the VDL Mode 4 Specific Services. The VSS user could be higher layers in the VDL Mode 4 SARPs or an external application using VDL Mode 4

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

3.1.4 Definition of bit order

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

a) bit order in each burst subfield shall be indicated by subscript numbers. Bit 1 shall indicate the least significant bit; and

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

3.2 Abbreviations

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

A/A	Air/Air communications
A/G	Air/Ground communications
ACK	ACKnowledgement (burst)
ADS-B	Automatic Dependent Surveillance-Broadcast
AIRSAW	AIRborne Situational AWareness
AM(R)S	Aeronautical Mobile (Route) Service
A-SMGCS	Advanced Surface Movement Guidance and Control Systems
ATN	Aeronautical Telecommunication Network
ATS	Air traffic Services
BITE	Built-In Test Equipment
BND	Big Negative Dither
CCI	Co-Channel Interference
CDTI	Cockpit Display of Traffic Information
CPR	Compact Position Reporting
CRC	Cyclic Redundancy Code
CTRL	ConTRoL (DLPDU)
dB	deciBel
DLE	Data Link Entity
DLPDU	Data Link Protocol Data Unit
DLS	Data Link Service
DOS	Directory Of Services
erid	extended reservation ID
EUROCAE	EURopean Organization for Civil Aviation Equipment
FIS-B	Flight Information Service-Broadcast
FOM	Figure Of Merit
G/G	Ground-Ground communications
GFSK	Gaussian Filtered frequency Shift Keying
GNS-B	GNSS Augmentation Service-Broadcast
GNSS	Global aeronautical Navigation Satellite System
GSC	Global Signalling Channel
hex	hexadecimal
ICAO	International Civil Aviation Organization
ID	IDentity
INFO	INFOrmation (DLPDU)
ISO	International Organization for Standardization
LCI	Logical Channel Identifier
LME	Link Management Entity
MAC	Media Access Control
MEDUP	MEDiterranean Update Programme
MOPS	Minimum Operational Performance Specification
ms	milliseconds
NEAN	North European ADS-B Network
NIC	Navigation Integrity Category
NM	Nautical Mile
ns	nanoseconds
NSCOP	Negotiated Set Up Connection-Orientated Protocol
NUP	NEAN Update Programme
OSI	Open Systems Interconnection
PCO	Point of Control and Observation
PECT	Peer Entity Contact Table
QoS	Quality of Service
RF	Radio Frequency
RFI	Radio Frequency Interference
rid	reservation ID

RNP	Required Navigation Performance
RTS	Request To Send (DLPDU)
SAR	Search And Rescue
SARPs	Standards And Recommended Practices
SNAcP	Sub-Network Access control Protocol
SNDCF	SubNetwork Dependent Convergence Function
STDMA	Self-organizing Time Division Multiple Access
TCP	Trajectory Change Point
TDMA	Time Division Multiple Access
TIS-B	Traffic Information Service-Broadcast
TTCN	Tree and Tabular Combined Notation
UCTRL	Unacknowledged ConTRoL data broadcast (DLPDU)
UTC	Universal Time Coordinated
VDL	VHF Digital Link
VDR	VHF Digital Radio
VHF	Very High Frequency
VME	VDL Management Entity
VSS	VDL mode 4 Specific Services
ZOCOP	Zero-Overhead Connection-Oriented Protocol

4 General description of VDL Mode 4 mobile station link layer

4.1 General

4.1.1 Overview of VDL Mode 4

VDL Mode 4 is a VHF data link, providing digital communications between mobile stations (aircraft and airport surface vehicles) and between mobile stations and fixed ground stations. It is developed for Communications, Navigation, Surveillance (CNS)/Air Traffic Management (ATM) aviation applications, including broadcast applications (e.g. ADS-B) and point-to-point communications (e.g. ADS-C, CPDLC). VDL Mode 4 protocols support ADS-B and similar broadcast applications through the broadcast of short repetitive messages, with graceful adaptation to increasing traffic loads. Military use of VDL Mode 4 is possible if military aircraft are fitted with relevant equipment; this will further improve digital communications.

VDL Mode 4 transmits digital data in a standard 25 kHz VHF communications channel and divides the communication channel into a large number of *time slots*. The start of each slot is an opportunity for a station to transmit.

VDL Mode 4 is built on the Self-organizing Time Division Multiple Access (STDMA) concept, in which the time-slots are synchronized to UTC-time, and stations advertise their intention to transmit in a specified time-slot by means of a reservation protocol carried in a prior transmission. For convenience, a group of contiguous time slots spanning a period of 60 s is termed a *superframe*. Each time slot may be used by a station for transmission of data. The exact timing of the slots and planned use of them for transmissions are known to all users in range of each other, so that efficient use of the data link can be made and users do not transmit simultaneously. As a result of this self-organizing protocol, VDL Mode 4 is capable of operating outside the coverage of a ground infrastructure and can therefore support air-air as well as ground-air data communications and applications. Under some circumstances, in e.g. high density airspace, a ground infrastructure may be used to manage the system to further improve overall performance.

In most respects, the VDL Mode 4 mobile station follows the provisions of the ICAO standards material for VDL Mode 4. Within the ICAO standard, there are some requirements which apply explicitly only to ground stations. 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 this clause.

4.1.2 Relationship to OSI reference model

The VDL Mode 4 sub-system implements the three lower layers of the OSI model as illustrated in figure 4.1.

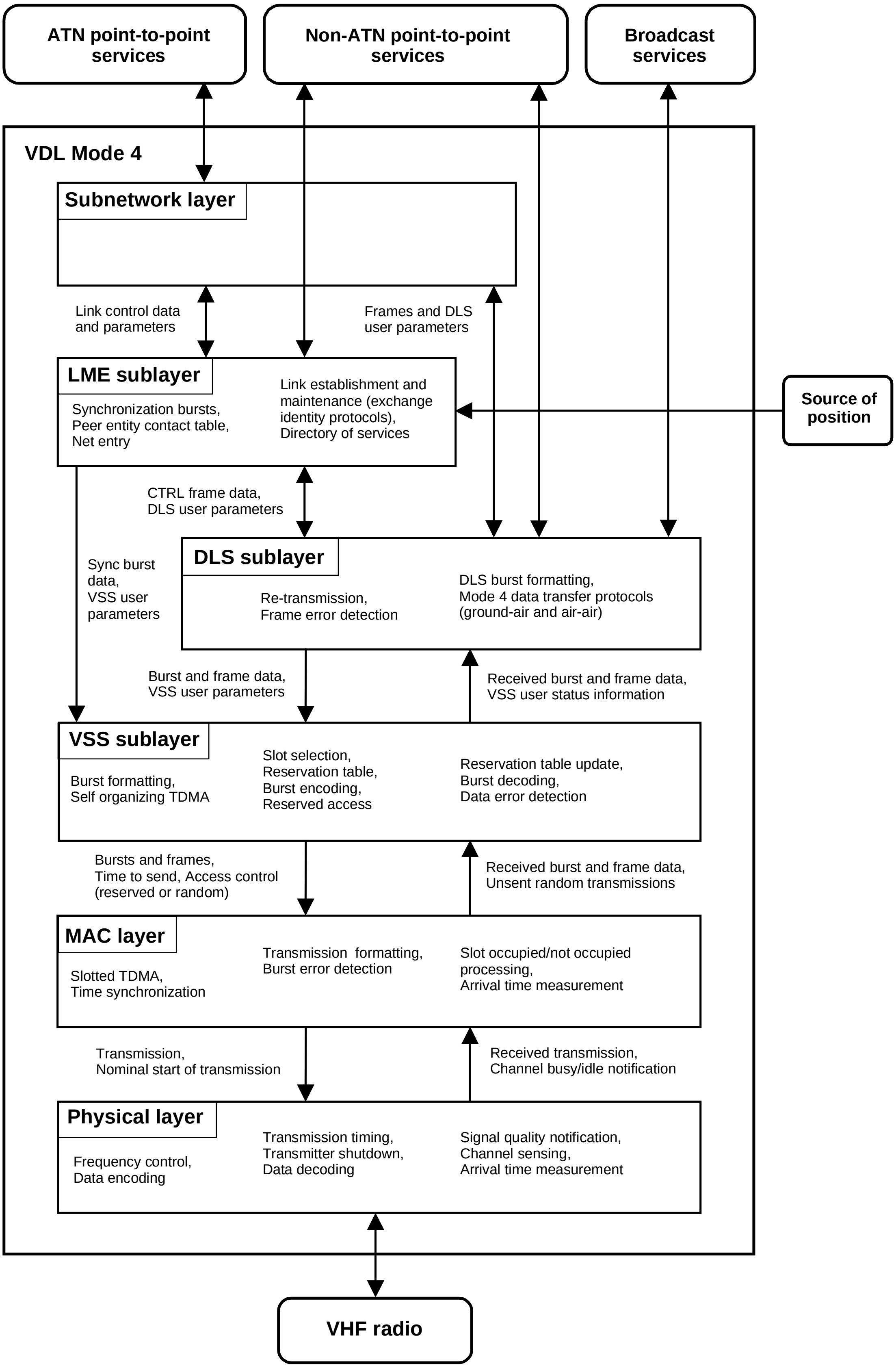

Figure 4.1: Layered structure of VDL Mode 4

Layer 1 (Physical layer): provides transceiver frequency control, bit exchanges over the radio media, and notification functions. These functions are more often known as "radio" and "modulation" functions.

The ICAO VDL SARPs defines the physical layer for VDL Mode 4: The modulation scheme is Gaussian Filtered Frequency Shift Keying (GFSK), at a nominal bit rate of 19 200 bits/second.

Layer 2 (Link Layer): is split into three sublayers and a management entity:

- The Media Access Control (MAC) sublayer provides access to the Physical layer by a simple Time Division Multiple Access (TDMA) algorithm under the control of the next higher sublayer. It also provides system time functions to co-ordinate the TDMA channel access.
- The VDL Mode 4 specific services (VSS) sublayer provides control of channel access using a self-organizing mechanism. The VSS also supports a number of ground controlled access protocols. The basic services are built on reserved, random and fixed access to the TDMA slots and support broadcast and point-to-point communication.
- The Data Link Services (DLS) sublayer performs frame exchanges, frame processing and error detection. The DLS protocols are adapted to make best use of the unique VSS channel access protocols.
- The Link Management Entity (LME) is in charge of the links between peer DLS sublayers and also the maintenance of the broadcast link functions.

Layer 3: the VDL SARPs defines only the lowest network sublayer of layer 3 (SNAcP). It is compliant with the subnetwork sublayer requirements defined in the ICAO ATN SARPs [8]. It provides packet exchanges over a virtual circuit, error recovery, connection flow control, packet fragmentation, and subnetwork connection management functions.

NOTE: The present document is limited to the core link layer specifications for VDL Mode 4 and hence does not provide specifications for the whole of layer 2, and nor does it specify any of layer 3.

4.1.3 VDL Mode 4 services

VDL Mode 4 supports two different types of communication services:

- VDL Mode 4 broadcast services;
- VDL Mode 4 point-to-point services.

NOTE: Other networks could also be supported but these have not been the focus of the ETSI VDL Mode 4 standardization work that has taken place so far. Some pre-operational implementations in Europe have been based on TCP/IP.

The VDL Mode 4 specific services include air-to-air, air-to-ground and ground-ground broadcast, and ground-to-air point-to-point (addressed) communications (for link control), with a minimum of overhead information for exchange of data including time-critical data.

In addition, VDL Mode 4 is intended to operate as an ATN sub-network and to support ATN compliant air-ground data communication services, employing point-to-point links involving the DLS function.

These VDL Mode 4 services are expected to be accommodated on multiple VHF channels. While data link channels is assumed to be separated from those supporting broadcast services, various broadcast functions and applications could share a channel. The possibilities for channel sharing depend on various constraints such as traffic densities, channel availability, certification requirements and ATS regulations, and may also differ between states and regions.

4.1.4 ADS-B function

The ADS-B function uses the VDL Mode 4 synchronization burst message formats to broadcast periodically an aircraft or vehicle's identity, position, altitude, time, intent and vector information for use by other mobiles and ground stations. Because position reporting is an integral part of communications management in VDL Mode 4, the core elements of ADS-B are already present on the link.

The size of the time slots on the data link is adapted to accommodate an ADS-B report in a transmission known as a synchronization burst. The *fixed part* of a synchronization burst contains core ADS-B information such as identity, a station's position altitude and time, but a synchronization burst may also accommodate additional ADS-B information in the *variable part* of the synchronization burst.

ADS-B supports many mobile-mobile surveillance applications such as Cockpit Display of Traffic Information (CDTI) (see note), Airborne Situational Awareness (AIRSAW), airborne separation, station-keeping and airport surface applications. When the VDL Mode 4 system also includes ground stations it is also able to support applications such as Advanced Surface Movement Guidance and Control Systems (A-SMGCS), Runway Incursion Prevention, enhanced ATC, Search And Rescue (SAR) co-ordination, etc.

NOTE: CDTI in this context means the functional capability to display position information, not the physical unit.

4.1.5 Operational scenarios

Three basic operational scenarios for VDL Mode 4 have been identified:

a) Autonomous operation is defined as the situation where no VDL Mode 4 ground infrastructure exists. Surveillance by means of ADS-B and air-air communication can take place between any users with overlapping cells by means of the self-organizing protocol, using autonomous transmissions. Applications supported by VDL Mode 4 will use the two Global Signalling Channels (GSCs) and possibly other channels as appropriate.

b) Single Cell operation is defined as the situation where overlapping VDL Mode 4 ground stations exist, but do not co-ordinate their operation over a ground network. Thus each single cell within the coverage of one ground station can be seen as an independently-operating system. Additional local channels may be available (e.g. to support SMGCS) and channel management can be supported by the transmission of the Directory of Service (DoS) message on the GSCs. In this scenario, stations may make autonomous transmissions, or else may be directed to transmit on a particular frequency and/or in specified slots by a ground station.

c) Multi Cell operation is defined as the situation where VDL Mode 4 ground stations co-ordinate their operation by means of ground networks. The number of VDL Mode 4 ground stations participating in multi cell operation affects overall system capacity and redundancy. Stations may make autonomous transmissions, or else may be directed by a ground station.

4.1.6 VDL Mode 4 fundamentals

VDL Mode 4 operation is built up from the following fundamental features which support ADS-B operation:

- A robust modulation scheme for encoding of data in each slot. VDL Mode 4 supports Gaussian Filtered Frequency Shift Keying (GFSK) with a transmission rate of 19 200 bits/second.
- A Self-organizing Time Division Multiplex access (STDMA) structure. In VDL Mode 4, channel time is divided into fixed length time slots. A superframe consists of a group of slots that span a period of 60 s and contains 4 500 slots (equivalent to 75 slots per second).
- A timing reference providing a unique marker for the start of each communications slot. The timing concept used in VDL Mode 4 is based upon Universal Co-ordinated Time (UTC). In the event that a station loses its primary source of UTC time, it may resort to a failure mode known as secondary timing with reduced precision. A possible source of secondary time may be derived from the time of arrival of synchronization bursts received from another station declaring primary time.

NOTE 1: The timing source is typically GNSS, but other sources may be used as long as they can be related to UTC.

- Position information from the aircraft's navigation system or internal GNSS unit is used to organize access to the slots. If a station loses its source of position information it may continue to derive position from the time of arrival of synchronization bursts received from other stations advertising certified data quality. This is known as secondary navigation. Stations operating on secondary timing do not offer certified data quality and thus cannot be used for secondary navigation.

NOTE 2: Requirements relating to the use of secondary navigation are outside the scope of the present document.

- A flexible message structure that can support a wide range of broadcast and data transfer protocols.
- A slot selection function that determines when a station can access the channel and maintains information on the current and planned slot assignments.
- A slot access management function, controlling the use of each slot.
- A number of link management functions that support access to data link services on a wide range of channels.

4.1.7 Possible configuration of airborne equipment

It is not the intention of the present document to prescribe a particular physical architecture for the VDL Mode 4 transceiver. It is assumed that the equipment will include all the relevant functionality defined by ICAO SARPs, as detailed in clause 5 of the present document, but that additional supporting functions such as determination of position and other air derived data will be performed external to the VDL Mode 4 transceiver. However, other architectures may be more appropriate to meet user requirements.

To meet the provisions of the present document, it is required that the equipment is tested in conjunction with all the physical units involved in the implementation of the functionality specified in clause 5, including the provision of the time reference, but excluding the derivation of data for transmission over the air-ground link. Where necessary, appropriate Points of Control and Observation (PCO) must be provided internally to the equipment to allow the tests specified in the present document to be performed.

It may be expected that VDL Mode 4 transceivers will be fitted to a wide range of aircraft configurations, each having differing requirements in terms of the services to be supported by the equipment and tolerance to equipment failure. In order to reflect such differing requirements, the following guidance is offered on the equipment configurations expected to be required to meet operational requirements. Other equipment configurations are not excluded, but manufacturers will be required to demonstrate by supporting analysis that an alternative configuration is capable of meeting the appropriate operational requirements.

For Air Transport applications, a dual or triple installation of VDL 4 transceivers is foreseen, each consisting of three or more receivers, all capable of simultaneous operation on independent frequencies, together with a frequency agile transmitter. For less sophisticated aircraft, including General Aviation, a transceiver consisting of only two receivers and a single transmitter may be suitable. However, such an installation will restrict the VDL Mode 4 applications and services capable of being supported, and will provide lower system availability.

No single transceiver will be required to transmit simultaneously on two different frequencies or be required to receive whilst transmitting on any other channel in the same VHF Digital Radio (VDR).

A discussion of possible aircraft architectures is provided in the Eurocontrol VDL Mode 4 Airborne Architecture study [12] where it is assumed that VDL Mode 4 functionality is located in the VDR with external functional interfaces providing:

- VHF antenna;
- UTC second timing;
- position/velocity/time;
- surveillance data management, including data required for broadcast services;
- data communications management, including connection to the Aircraft data subnetwork;
- connections between VDL Mode 4 functions in different VDRs (these links are discussed in clause 4.1.9).

A possible aircraft architecture is illustrated for guidance in figure 4.2. However it should be noted that the final architecture will be up to the industry.

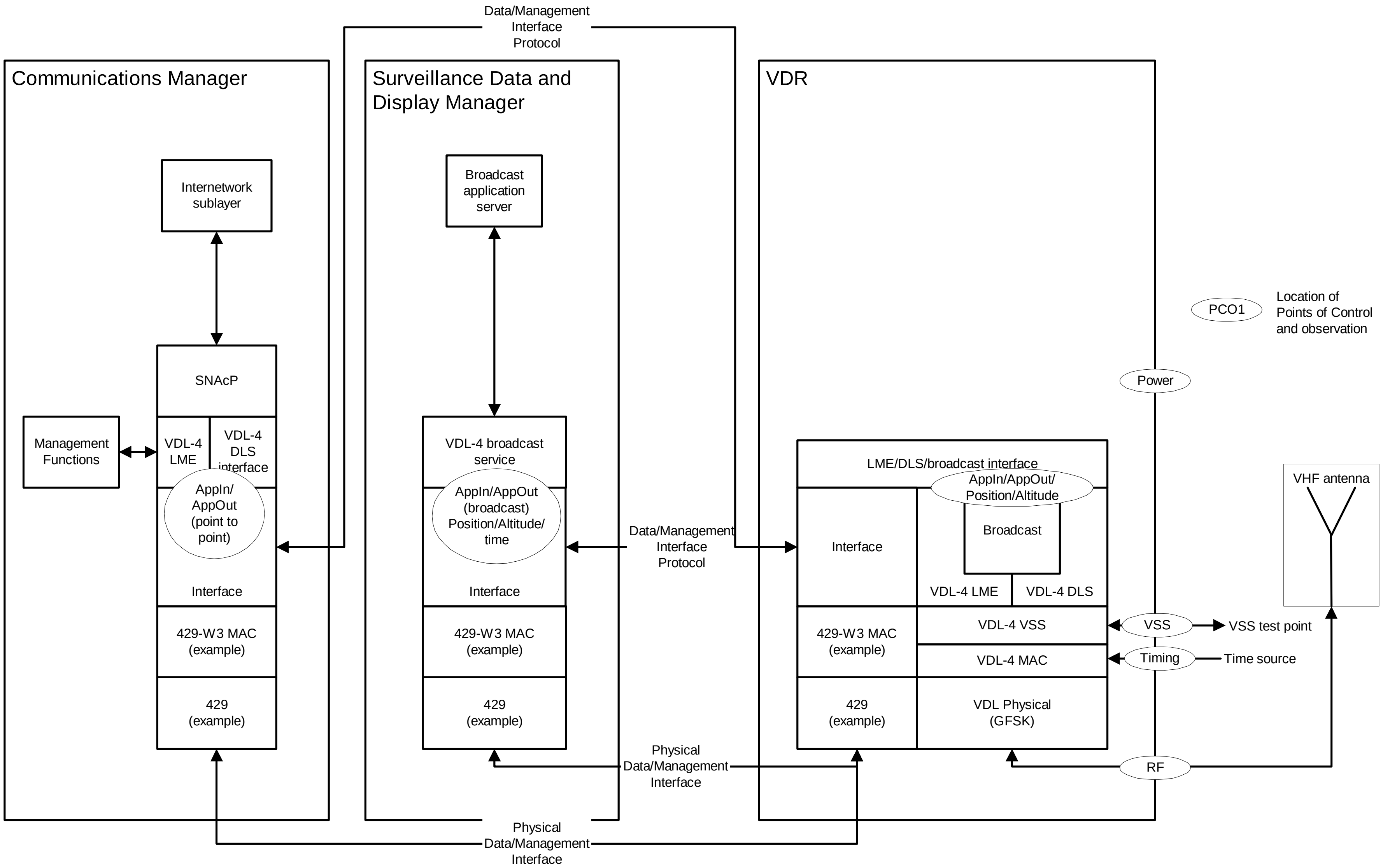

Figure 4.2: Possible VDL Mode 4 aircraft architecture

The VDR interfaces via an appropriate interworking protocol such as ARINC 429 to:

- a surveillance data manager, providing broadcast data including position, velocity and time;
- a communication manager, providing connection to the subnetwork.

A direct source of UTC second timing is provided via a separate link.

NOTE: It is assumed that the VDL Mode 4 requirement for a precise UTC timing input would not be met using an ARINC 429 connection, even though ARINC 429 is capable of distributing time information.

Figure 4.2 also illustrates the location of points of control and observation (these are discussed in more detail in clause 7.2).

4.1.8 Classes of equipment

Two types of transmitter are specified in the present document (see clause 6.2.3 of EN 302 842-1 [4]) designed to meet operational ranges with margin as specified in the table below.

	Air-Air	Air-Ground
Class A	> 120 NM	> 160 NM
Class B	> 60(50) NM (B-B)	> Horizon range at 12 000 ft

Air transport aircraft shall use a Class A transmitter.

Recommendation:

For other aircraft/vehicles, manufacturers should specify a Class A transmitter, wherever practicable.

For small aircraft (e.g. light general aviation and gliders) or ground vehicles, a reduced power Class B transmitter may be used, capable of supporting a restricted range ADS-B service, when considerations of cost, size, weight, or power consumption prevent implementation of a Class A transmitter.

4.1.9 Transceiver cross-links

Access to the widest range of VDL Mode 4 applications, and the highest tolerance to failure will be provided by a multiple installation of VDL Mode 4 transceivers with cross-links for the exchange of information among the transceivers (see clause 6.8.3). Such cross-links are intended to support exchange of reservation information between transceivers to allow one transceiver to continue operation on a channel previously monitored by the other transceiver under failure conditions, as well as to support co-ordination of channels and applications supported by each transceiver in normal operation.

Information transferred by such cross-links may also be required to co-ordinate the selection of slots for transmission by a transceiver, so as to avoid interference with reception by another transceiver on the aircraft, in the event that adequate RF isolation between transceivers cannot be achieved.

The behaviour and testing of cross-links between VDL Mode 4 transceivers is outside the scope of the current version of the present document.

NOTE: The use of cross-links is very dependent on the airborne architecture. The present document focuses on the minimum equipment, which is a single box supporting two channels. In the event that cross-links are used, further requirements and tests may need to be defined.

4.1.10 Overall structure of specifications for VDL Mode 4

The specifications for VDL Mode 4 mobile stations are split into four volumes as illustrated in figure 4.3.

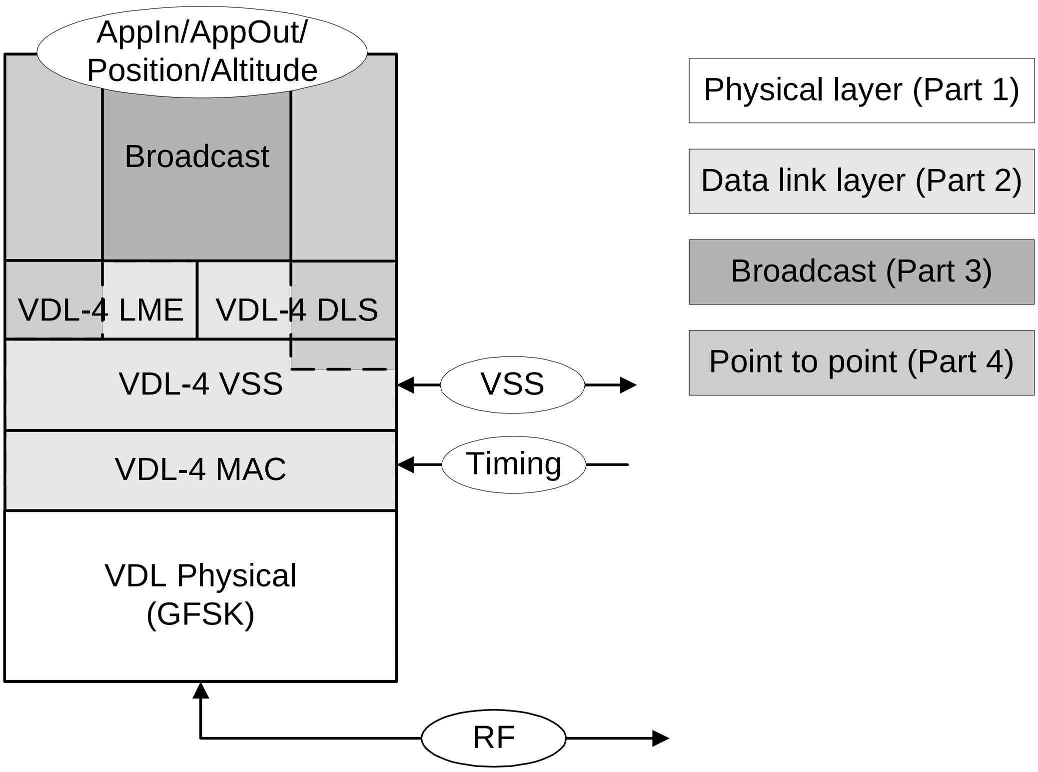

Figure 4.3: Structure of VDL Mode 4 mobile station specifications

It is assumed that:

- An equipment providing broadcast services only would conform to the requirements of parts 1, 2 and 3.
- An equipment providing point-to-point services only would conform to the requirements of parts 1, 2 and 4.
- An equipment providing all services would conform to the requirements of parts 1, 2, 3 and 4.

Part 1 (Physical layer - EN 302 842-1) [4] provides the functions necessary to establish a physical layer link between stations.

Part 2 (Data link layer - the present document) provides the functions necessary to:

- Establish a stream of broadcast transmissions protected by broadcast reservation types.
- Respond correctly to all reservation types.
- Provide repetitive transmission of aircraft position.
- Provide link management services to enable a ground station to control quality of service parameters via ground broadcast transmissions.

The present document is based on ICAO VDL4 Technical Manual [1] and includes:

- **All functions associated with the MAC sublayer**.
- **All functions associated with broadcast services within the VSS sublayer**.
- **All functions associated with the receipt of point-to-point reservation types within the VSS sublayer**.
- **Station addresses and broadcast services from the DLS sublayer**. Note that the present document provides broadcast services via single unacknowledged transmissions. The present document specifically does not implement long transmission procedures as these apply only to point-to-point communication.

- **LME functions to support the repetitive broadcast of position within a synchronization burst**. This includes the encoding of the fixed part of the synchronization burst and a variable part containing no further information. This is sufficient to support communication management but is supplemented in EN 302 842-3 [9] with a greater range of information to support broadcast services. The core also supports a simple synchronization burst request message making it possible for a communications manager to establish periodic reporting streams. This is again supplemented in EN 302 842-3 [9] to support a greater level of control by a ground station over the rate and content of synchronization bursts.

- **Compact Position Reporting (CPR) encoding to support the fixed part of the synchronization burst**.

In the VSS layer, the present document excludes:

- Information transfer protocol transmission procedures (see EN 302 842-4 [10]). Note that the reception procedures for this protocol are included in the present document to support interoperability between broadcast only and point-to-point only equipment.

Part 3 (EN 302 842-3 [9]) (broadcast services) defines messages and additional protocols to support ADS-B, TIS-B, FIS-B and GNS-B. The main purpose of EN 302 842-3 [9] is to define message formats which are transmitted and received using the broadcast services of the DLS defined within the present document. In addition, some additional protocols are added within the ADS-B specifications.

EN 302 842-3 [9] specifications are based on:

- ICAO Technical Manual specifications for ADS-B and CPR offset encoding.

- Specifications for TIS-B, FIS-B and GNSS Augmentation adapted from material developed as part of European Commission sponsored NUP and MEDUP trials of VDL Mode 4 equipment, and ICAO GNSS SARPs.

Part 4 (EN 302 842-4 [10]) (point-to-point services) provides air-to-ground and air-to-air point-to-point services based on the ICAO VDL4 Technical Manual [1]. EN 302 842-4 [10] includes:

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

- Point-to-point link control within the LME.

Note that EN 302 842-4 [10] covers the establishment, termination and handover of links between ground and mobile stations and ground station coordination. Decisions to establish, terminate or handover links between ground stations are local issues and beyond the scope of these specifications.

EN 301 842 [15] covers all VDL Mode 4 services applicable to ground-based equipment including broadcast and point-to-point communications.

4.1.11 Equipment performance verification

To test the equipment for compliance with the performance requirements, three types of test are specified:

- bench tests;

- environmental tests;

- installed equipment tests.

The performance requirements for each type of test and the corresponding test procedures are specified in the present document. The order of test suggests that the mobile station be subjected to a succession of tests as it moves from design into design validation and equipment qualification. The objectives of these tests are described below.

Bench tests

The equipment will be subjected to bench test to verify compliance with the performance requirements under a controlled environment. The test results may be used as the basis for approval of equipment design, equipment qualification, and acceptance. The bench test procedures are specified in clause 7.

Environmental tests

Upon successful completion of bench tests, the equipment will be subject to environmental tests to verify compliance to the performance requirements under extreme environmental conditions expected in actual operations and abnormal conditions. The test results may be applied to equipment qualification and acceptance. The environmental tests are specified in clause 8 of EN 302 842-1 [4].

Installed equipment tests

After successful completion of the environmental tests, the equipment will be installed in the aircraft for further testing to verify compliance with the performance specifications of EN 302 842-1 [4] clause 8 in the aircraft environment. The installed equipment tests will be conducted primarily with the aircraft on the ground using simulated or operational system inputs and may be supplemented with flight tests. The test results may be used to isolate design faults that are traceable to the general aircraft environment, e.g. Radio Frequency Interference (RFI) and the aircraft power supply system. As in the case of bench and environmental tests, installed equipment tests are critical to equipment qualification and acceptance. The installed equipment tests are specified in clause 6.24 to 6.26.

5 Minimum performance specification under standard test conditions

5.1 MAC sublayer

5.1.1 Services

Requirement reference	
5.1.1.1	The MAC sublayer shall acquire the shared communication path so as to provide the services defined in clause 5.1.2.
NOTE: The functions performed by the MAC sub-layer should be "transparent" to higher functional layers.	

5.1.2 MAC sublayer services

Requirement reference	
5.1.2.1	The MAC sublayer shall accept from the physical layer a continuous indication of channel idle/busy status and signal level (see clause 5.1.5).
5.1.2.2	The MAC sublayer shall accept from the VSS sublayer a burst for transmission, accompanied by the time to transmit it.
5.1.2.3	The MAC sublayer shall provide to the VSS sublayer the received burst data, slot busy/idle status, slot occupancy status, signal level and the status of bursts sent for transmission.

5.1.3 MAC sublayer parameters

5.1.3.1 General

Requirement reference	
5.1.3.1.1	MAC service system parameters shall be as described in table 5.1.

Table 5.1: MAC service system parameters

Symbol	Parameter Name	Minimum	Maximum	Default	Increment
M1	Number of slots per superframe	n/a	n/a	4 500 slots	n/a

5.1.3.2 Parameter M1 (number of slots per superframe)

Requirement reference	
5.1.3.2.1	The parameter M1 shall be the number of available slots per superframe.
5.1.3.2.2	A superframe shall span a period of 60 s.
NOTE: M1/60 slots span a time interval of one second. The M1 increment forces M1/60 to be an integer. This simplifies the protocol since a slot counter (or equivalent) can be started at the boundary between any two consecutive UTC seconds.	

5.1.3.3 Parameter M2_inc and M2_limit (MAC layer control parameters for network entry)

Requirement reference	
5.1.3.3.1	An m2 filter shall be maintained by a station for each active channel as a measure of the uncertainty of the reservation data.
5.1.3.3.2	Each m2 filter is controlled by parameters M2_inc and M2_limit, which, in turn, define the parameters of the following algorithm in which m2 is updated after every slot: $m_{2k}+1 = m_{2k} + M2\_inc$ if receiver function on the channel is either: a) blocked by own station transmission on another channel (e.g. through a common antenna); or b) disabled due to failure of the VDL4 equipment; or c) disabled due to equipment power failure; or d) any other inability to receive. $m2_{k+1} = \max[(m2k - 1), 0]$ if receiver function is not blocked
5.1.3.3.3	If m2 ≥ M2_limit, m2 shall be reset to zero (m2 = 0) and the station shall execute a network entry procedure.
NOTE 1: For a station that is able to make reservations on a channel as defined in clause 5.2.6.1, the m2 (measure of the uncertainty of the reservation data) filter controls the need to execute a network entry procedure because of the temporary loss of receiving function on the channel. The loss of receiving function may be due to a number of causes including known co-site transmissions, failure of the VDL4 equipment or equipment power failure. NOTE 2: The objective of parameters M2_inc and M2_limit is to force re-entry when the reservation table may have been corrupted by a lack of recent reservation data, but not requiring re-entry due to minor gaps in reception that may have been caused by limited and isolated transmissions on other channels. Parameters M2_limit and M2_inc can be reset by ground station command on a per channel basis.	

5.1.3.4 Inspection of m2 filter following transceiver failure

Requirement reference	
5.1.3.4.1	Immediately following power interrupt or failure affecting a VDL Mode 4 transceiver with duration exceeding that defined as a short power interrupt in clause 6.3.5 of EN 302 842-1 [4], the transceiver shall: a) ensure that full operation has returned; b) verify that the m2 filter can be accurately updated according to the period of down-time that has occurred (see note); c) increment the m2 filter according to the period of down-time that has occurred, according to the requirement in table 3-1A with ICAO Technical Manual reference 1.2.2.2b; d) inspect the updated value of the m2 filter: if m2 is greater than or equal to M2limit, m2 shall be re-set to 0, and the station shall execute a network entry procedure.
NOTE: It is assumed that the number of slots that have passed during the period of equipment failure can be calculated from comparison of the current time (obtained following recovery) with the time at which failure started (assumed to be recorded).	

5.1.4 Time synchronization

5.1.4.1 Primary

Requirement reference	
5.1.4.1.1	Under normal operating conditions, a station shall maintain time synchronization such that the start of each successive group of M1/60 slots is synchronized with the start of any Universal Time Coordinated (UTC) second to within a two-sigma value of 400 ns.

5.1.4.2 Secondary

Requirement reference	
5.1.4.2.1	A station shall be capable of maintaining time synchronization such that the start of each successive group of M1/60 slots is synchronized with the start of any UTC second to within a two-sigma value of 15 µs.
5.1.4.2.2	Only when the primary source fails shall secondary time be used.
5.1.4.2.3	A station using secondary time shall revert to primary time whenever primary time is available.
5.1.4.2.4	A station that is unable to support either primary or secondary time shall not transmit on any VDL Mode 4 frequency.
5.1.4.2.5	A station which regains primary or secondary time after a period not maintaining either primary or secondary time shall be required to re-enter the network using an appropriate procedure as defined in clause 5.4.4.3.
NOTE 1: One method of obtaining secondary synchronization mode is to synchronize to the slot boundaries that are defined by a station declaring primary time. NOTE 2: Secondary time is used only when the primary source has failed. A station using secondary time shall however revert to primary time whenever primary time is available. NOTE 3: Secondary time is regarded as failure mode. NOTE 4: Tertiary time, as described by the ICAO VDL4 Technical Manual [1], has not been included as a requirement in the present document due to the opinion of the ETSI Working Group that insufficient evidence was available to be able to verify correct operation of a "floating network" of stations operating on tertiary time.	

5.1.4.3 Alignment to UTC second

Requirement reference	
5.1.4.3.1	For stations maintaining primary or secondary time, the start of each successive group of M1/60 slots shall be aligned with a UTC second.

5.1.4.4 Data quality level

Requirement reference	
5.1.4.4.1	The certified quality level shall indicate that timing and position information provided by the station can be used by other stations as a means of deriving position information.
5.1.4.4.1a	When a station is deriving position information from the transmissions of other stations, it shall only use data from stations that have declared the certified data quality level.
5.1.4.4.2	The secondary timing level shall not indicate the certified quality level.
5.1.4.4.3	A station shall not declare the certified data quality level (see note 2).
NOTE 1: When, in future, the criteria for supporting secondary navigation are established, this requirement may be removed. NOTE 2: The certification of stations for use as "pseudolites" in order to support secondary navigation will be under the control of an authority, such as the civil aviation administration.	

5.1.5 Slot idle/busy notification

5.1.5.1 Slot idle detection

Requirement reference	
5.1.5.1.1	A station shall consider the slot idle if the channel idle/busy status supplied by the physical layer is idle at the start of the slot.

5.1.5.2 Slot busy detection

Requirement reference	
5.1.5.2.1	A station shall consider the slot busy if the channel idle/busy status is busy at the start of the slot.

5.1.5.3 Slot occupied detection

Requirement reference	
5.1.5.3.1	A slot shall be considered occupied if the channel is considered to be continuously busy for a period of at least 5 ms during the slot.
NOTE: The slot occupied detection is used to monitor the operations of peer stations and to provide an indication that there might be transmissions in a slot even if those transmissions cannot be decoded by the MAC layer. This is different from the slot idle/busy state, which affects in part the station's ability to make a random transmission.	

5.1.5.4 Signal level indication

Requirement reference	
5.1.5.4.1	The MAC sublayer shall accept from the physical layer an indication of the signal level.
NOTE: The signal level indication is used in the periodic broadcast protocol as defined in clause 5.2.10. The measurement is for relative purposes only and need not be calibrated to any standard.	

5.1.6 Transmission processing

Requirement reference	
5.1.6.1	Bursts received from the VSS sublayer shall be forwarded to the physical layer, together with the time for transmission.
5.1.6.2	A station shall begin transmissions only at the beginning of the slot boundary as determined by its local clock, except if the transmission is a "delayed burst".
5.1.6.3	Delayed bursts shall begin 4 ms after the start of the slot boundary, if the slot is idle at that point.
NOTE: The delay allows time for other stations to begin transmitter ramp up, for the signal to travel the propagation distance, and for the slot busy detector to determine the appearance of a signal. A delayed burst can fit in a single slot and thus preserve nominal propagation guard time, even if transmission begins late. The delay may be somewhat shorter but not longer; the 4 ms value is selected to ease design and ensure robustness of the slot busy detector.	

5.1.7 Received transmission processing

Requirement reference	
5.1.7.1	Bursts with an invalid Cyclic Redundancy Code (CRC) shall be discarded.
5.1.7.2	Bursts with valid CRCs shall be forwarded to the VSS sublayer, along with the received time of transmission and signal quality parameters.

5.2 VSS sublayer

NOTE 1: There is one VSS sub-layer entity for each VDL Mode 4 channel that is accessed by the station. The VSS sub-layer provides services to the VDL Mode 4 management entity (VME) as well as to the LME associated with other VDL Mode 4 peer systems, their associated Data Link Entities (DLEs) and the DLS. The VSS is served by the MAC that is associated with its particular VDL Mode 4 channel. NOTE 2: This clause describes the services provided by bursts as well as some sample protocols and procedures which may be amended, extended or ignored by any specific burst application. NOTE 3: Other protocols may be defined for unique applications; however, it is expected that most bursts will use one of the protocols in clause 5.2.1. It should be noted that the various reservation fields cannot be redefined for the protocols in clause 5.2.1.

5.2.1 Services

5.2.1a Multiple access

Requirement reference	
5.2.1a.1	The VSS sub-layer implements protocols that enable all stations to transmit while at the same time, maintaining high system throughput, low transit delays and low probability of collisions. These protocols shall include: a) reserved access (see clause 5.2.6); b) null reservation (see clause 5.2.9); c) periodic broadcast (see clause 5.2.10); d) incremental broadcast (see clause 5.2.11); e) combined periodic broadcast and incremental broadcast (see clause 5.2.12); f) Big Negative Dither (BND) broadcast (see clause 5.2.13); g) unicast request (see clause 5.2.14); h) information transfer request (see clause 5.2.15); i) directed request (see clause 5.2.16); j) block reservation (see clause 5.2.17); k) response (see clause 5.2.18); and l) random access (see clause 5.2.7).
NOTE: The designation of "reserved access", "random access" and "fixed access" apply to the protocols in the transmitting station, but not to any message indicator or flag in the transmitted frame or burst.	

5.2.1.1 Error detection

Requirement reference	
5.2.1.1.1	The VSS sublayer shall compute a 16 bit CRC according to ISO/IEC 13239 [3] to facilitate detection by the MAC sublayer (see clause 5.1.7) of data corruption during transmission.

5.2.1.2 Channel congestion

Requirement reference	
5.2.1.2.1	The VSS sublayer shall notify the LME sublayer whenever channel congestion is detected (see clauses 5.2.7.2.2 to 5.2.7.2.5).

5.2.2 Burst format

5.2.2.1 VSS burst structure

Requirement reference	
5.2.2.1.1	VSS bursts shall conform to ISO/IEC 13239 [3] frame structure except as specified in table 5.2.
5.2.2.1.2	Bits denoted "res" shall be set to zero on transmit and ignored on receipt.
	Recommendation
5.2.2.1.2a	Eight bits should be allowed for bit stuffing for the first slot, and for each additional slot that the burst occupies.
5.2.2.1.2b	If, after bit stuffing, the burst to be transmitted is longer than the number of slots allocated for the transmission, then the burst shall not be transmitted and the VSS user notified of the failure to transmit.
NOTE 1: The maximum burst length is defined in clause 5.2.3.5. NOTE 2: A burst occupying a single slot has 24 octets of data. Thus, assigning 8 bits for bit stuffing and 2 octets for the flags, a maximum single-slot burst has a value of "n" equal to 21. Note that a burst can be up to VS5 slots in length. Bursts can consist of a single block of data between two flags, as illustrated in table 5.2, or can consist of a number of blocks of data with each block separated from the next by a flag. NOTE 3: Aeronautical Telecommunications Network (ATN) applications are supported by DLS bursts (see clause 5.3.2.3). NOTE 4: The lengths of the message ID (mi) and information (in) fields depend on the message type being used. NOTE 5: The lengths of the extended reservation ID (erid) and reservation data (rd) fields depend on the reservation protocol being used. The rd field may also be subdivided into further fields, depending on the reservation protocol.	

Table 5.2: Burst format

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
flag	-	0	1	1	1	1	1	1	0
autonomous/directed flag (a/d), reservation ID (rid), version number (ver)	1	s_{27}	s_{26}	s_{25}	ver_3	ver_2	ver_1	rid	a/d
source address (s)	2	s_{24}	s_{23}	s_{22}	s_{21}	s_{20}	s_{19}	s_{18}	s_{17}
	3	s_{16}	s_{15}	s_{14}	s_{13}	s_{12}	s_{11}	s_{10}	s_9
	4	s_8	s_7	s_6	s_5	s_4	s_3	s_2	s_1
message ID (mi)	5	mi_k	....			mi_4	mi_3	mi_2	mi_1
information	6	in_k	....						
	7 to n – 5	....							
	n – 4	....							
reservation data (rd)	n – 3		in_1	rd_k	....				
extended reservation ID (erid)	n – 2	$erid_k$	....			$erid_1$			rd_1
CRC (c)	n – 1	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
	n	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0
NOTE: "n" is the burst length expressed in octets.									

....	Denotes variable length field

5.2.2.2 Version number

Requirement reference	
5.2.2.2.1	The version number (ver) subfield shall indicate the version of VDL Mode 4 supported by the station.
5.2.2.2.2	It shall be set to 000 (see table 5.2).
5.2.2.2.3	If the station receives a burst in which the version number is non-zero, it shall inform the VSS user that a non-zero version number has been received and ignore the rest of the burst.
NOTE: "000" represents the three bits of the version field. See table 5.2.	

5.2.2.3 Source address

Requirement reference	
5.2.2.3.1	The source address (s) of the transmitting station shall be encoded in the 27-bit field as defined in table 5.2.
5.2.2.3.2	The address format shall be as defined in clause 5.3.1.3.

5.2.2.4 Message ID

Requirement reference	
5.2.2.4.1	The message ID (mi) of the burst shall be encoded in the variable length field as defined in table 5.2.
5.2.2.4.2	The bits of the burst message ID field shall be as defined in table 5.3.

Table 5.3: Message ID assignment

Message ID field								Assigned burst type	VSS user
mi_8	mi_7	mi_6	mi_5	mi_4	mi_3	mi_2	mi_1		
x	x	x	x	x	x	x	0	Synchronization burst (see clause 5.4.2)	LME
x	x	x	x	1	1	0	1	DLS bursts (see clause 5.3.1.3 of the present document and clause 5.1.2.2 of EN 302 842-4 [10])	DLS
x	x	x	x	x	0	1	1		
x	x	x	x	x	1	1	1		
1	0	0	1	0	1	0	1		
x	x	1	0	0	1	0	1		
x	x	x	x	x	0	0	1	General request burst	Defined by r-mi
x	0	0	0	0	1	0	1	No operation	VSS
x	1	0	0	0	1	0	1	Network entry burst	VSS
0	0	0	1	0	1	0	1	Message ID extension to next 4 bits	
x	1	1	1	0	1	0	1	General response burst	Defined by r-mi
x	0	1	1	0	1	0	1	Reserved for future use	
x	1	0	1	0	1	0	1	Reserved for future use	
NOTE: Bits denoted as "x" are available for use within the information field. "r-mi" is the requested message ID field.									

Requirement reference	
5.2.2.4.3	The message ID shall define the VSS user which is responsible for handling the message, following completion of processing required within the VSS.

5.2.2.5 Information field

Requirement reference	
5.2.2.5.1	The optional information field (in) shall contain VSS user defined data.
NOTE:	The content of the information field (in) is generally a matter for definition by the applications using VDL Mode 4. General request and general response message formats are specified in clause 5.2.19 and clause 5.2.20 respectively.

5.2.2.6 Reservation fields

Requirement reference	
5.2.2.6.1	The reservation ID (rid) of the burst shall be encoded in the 1-bit field as defined in table 5.2.
5.2.2.6.2	If the reservation ID equals 1, this shall indicate that the reservation type is either a null reservation (see clause 5.2.9), a periodic broadcast reservation (see clause 5.2.10) or a combined periodic broadcast and incremental broadcast reservation (see clause 5.2.12) and that there is no extended reservation ID (erid).
5.2.2.6.3	Otherwise, the extended reservation ID field shall indicate other reservation types as defined in table 5.4.
NOTE:	The rid subfield (and erid subfield, if present) defines the interpretation of the reservation data (rd) field.

Table 5.4: Extended reservation ID field (erid)

Extended reservation ID field (erid)					Reservation type
Octet n - 2					
Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	
0	0	0	0	0	Response burst (no reservation)
0	0	0	0	1	Big negative dither (BND) reservation
0	0	0	1	0	Superframe block reservation
0	0	0	1	1	Second frame block reservation
0	0	1	0	x	Unicast request reservation
0	0	1	1	0	Reserved for future allocation
to					
0	1	0	0	1	
0	1	0	1	0	Information transfer request reservation
0	1	0	1	1	Reserved for future allocation
0	1	1	0	0	Directed request reservation
0	1	1	0	1	Reserved for future allocation
to					
0	1	1	1	1	
1	0	x	x	x	Incremental broadcast reservation
1	1	0	0	0	Reserved for future allocation
to					
1	1	1	1	1	
NOTE: Bits denoted "x" are available for use within the reservation data (rd) field.					

5.2.2.7 Autonomous/directed flag

Requirement reference	
5.2.2.7.1	The autonomous/directed (a/d) flag shall be encoded as defined in table 5.5.
NOTE:	A station transmitting a burst with an a/d bit set to zero implies that it has a valid reservation table. A transmission by a mobile station that contains a periodic reservation and whose a/d bit is set to one will invoke the quarantining procedures described in clause 5.2.3.2.

Table 5.5: Autonomous/directed flag encoding

Subfield	Range	Encoding	Notes
autonomous/ directed (a/d)	Boolean	0 = random transmission or reserved transmission in a slot selected by this station. 1 = delayed burst transmission or reserved transmission in a slot selected by a peer station.	Identifies whether the station is transmitting based on its internal reservation table or if it is being directed by a peer.

5.2.3 VSS sublayer parameters

5.2.3.1 General

Requirement reference	
5.2.3.1.1	VSS service system parameters shall be as described in table 5.6.

Table 5.6: VSS sublayer parameters

Symbol	Parameter name	Minimum	Maximum	Default	Increment
VS1	Number of ground quarantined slots	0 slots	15 slots	4 slots	1 slot
VS2	Minimum CCI performance	6 dB	60 dB	12 dB	1 dB
VS4	Quarantine slot re-use range	0	1 000 NM	300 NM	10 NM
VS5	Maximum burst length	1 slot	16 slots	10 slots	1 slot

5.2.3.2 Parameter VS1 (number of ground quarantined slots)

Requirement reference	
5.2.3.2.1	The parameter VS1 shall define the number of ground quarantined slots.
5.2.3.2.2	Quarantined slots shall be slots which may not be used by a mobile station unless directed by a ground station.
5.2.3.2.3	Quarantined slots shall be established by a ground station or network of coordinated ground stations under the following circumstances: a) A mobile station, A, will not reserve a slot or transmit on the slot boundary of the VS1 slots after a slot which has been reserved by a ground station, B, using a periodic broadcast reservation or which has been reserved by a mobile, C, using a burst with the autonomous/directed bit set to 1 and a periodic broadcast reservation field, unless the station (B or C) that has reserved the slot is at a range greater than VS4 from station A. In the case that station (B or C) that has reserved the slot is at a range greater than VS4 from station A, then station A will consider the slot to be unreserved. b) If a mobile station receives a periodic broadcast burst with the periodic offset (po) subfield set to zero and the periodic timeout (pt) subfield set to zero, then it will maintain ground quarantine for the current slot and for M1 slots after the current slot if it had previously contained a reservation associated with the same stream. Ground quarantine behaviour for any other slots associated with the same stream will be cancelled. c) A mobile station, A, will not reserve a slot or transmit in slots which have been reserved by a ground station, B, or a mobile station, C, using a block reservation, unless the station (B or C) that has reserved the slot is at a range greater than VS4 from station A, in which case station A will consider the slot to be unreserved.

5.2.3.3 Parameter VS2 (minimum CCI performance)

Requirement reference	
5.2.3.3.1	The parameter VS2 shall be used to control the CCI conditions by which a station Y may transmit given that another station X has reserved the same slot.
5.2.3.3.2	In the case where a station X and Y transmit in the same slot and station X's transmission is directed to another station Z, CCI conditions shall be fulfilled (a transmission from station X will not interfere with the transmissions from station Y and Z) if the ratio defined below: $ratio = 10\log\left(\frac{dist(Y \mid Z)^2}{dist(X \mid Z)^2}\right)$ is greater than VS2, where dist(Y/Z) is the distance between station Y and Z and dist (X/Z) is the distance between station X and station Z.
NOTE: This condition is applied twice to establish that the transmission between aircraft A and C will not interfere with the transmission between aircraft B and D (as described in table 5.2.9).	

5.2.3.4 Parameter VS4 (quarantine slot re-use range)

Requirement reference	
5.2.3.4.1	The parameter VS4 shall be used to control the range at which a quarantined slot may be re-used by a distant station.

5.2.3.5 Parameter VS5 (maximum burst length)

Requirement reference	
5.2.3.5.1	The parameter VS5 shall define the maximum burst length in slots including flags and zero bits inserted for transparency.

5.2.4 VSS quality of service parameters

5.2.4.1 General

Requirement reference	
5.2.4.1.1	Every burst processed by the VSS sublayer for transmission shall be associated with the parameters defined in table 5.7.

Table 5.7: VSS quality of service system parameters

Symbol	Parameter Name		Minimum	Maximum	Default	Increment
Q1	Priority		0	15	11	1
Q2a	Slot selection range constraint for level 1		0	1 000 NM	150 NM	1 NM
Q2b	Slot selection range constraint for level 2		0	1 000 NM	150 NM	1 NM
Q2c	Slot selection range constraint for level 3		0	1 000 NM	0 NM	1 NM
Q2d	Slot selection range constraint for level 4		0	1 000 NM	300 NM	1 NM
Q3	Replace queued data		FALSE	TRUE	FALSE	--
Q4	Number of available slots		1	20	3	1
$Q5_{min}$	VSS re-transmission parameters	minimum	0 sec	20 sec	0 sec	13,33 milliseconds
$Q5_{max}$		maximum	1 sec	20 sec	5 sec	13,33 milliseconds
$Q5_{mult}$		multiplier	1 sec	2,5 sec	1 sec	0,01 sec
$Q5_{exp}$		exponent	1	2,5	1,5	0,01
$Q5_{num}$		number of attempts	1	15	4	1
$Q5_{wait}$		maximum time to wait for a reply	1 sec	120 sec	60 sec	1 sec

5.2.4.2 Parameter Q1 (priority)

Requirement reference	
5.2.4.2.1	The parameter Q1 shall be the priority of the transmission and shall be as defined in table 5.8.

Table 5.8: Priority levels

Message categories	Q1
Unassigned	15
Network/systems management	14
Distress communications	13
Urgent communications	12
High priority flight safety messages	11
Normal priority flight safety messages	10
Meteorological communications	9
Flight regularity communications	8
Aeronautical information service messages	7
Network/systems administration	6
Aeronautical administrative messages	5
Unassigned	4
Urgent priority administrative and UN charter communications	3
High priority administrative and state/government communications	2
Normal priority administrative	1
Low priority administrative	0
NOTE: Q1 = 15 is reserved for future use.	

5.2.4.3 Parameters Q2a to Q2d (slot selection range constraint for level n)

Requirement reference	
5.2.4.3.1	The parameters Q2a to Q2d shall be used to impose range constraints on the slot selection process for levels 1 to 4 defined by table 5.9.

Table 5.9: Slot selection criteria

	Selection conditions		
Selection priority	**Planned transmission by station A**	**Previously reserved transmission by station B**	**Minimum distance between station A and station B**
Level 0	Any	Unreserved	Not applicable
Level 1	Broadcast or CCI protected communication with station C	CCI protected communication with station D	Q2a
Level 2	Broadcast or CCI protected communication with station C	Broadcast	Q2b
Level 3	Broadcast or CCI protected communication with station C	Broadcast or CCI protected communication with station D	Q2c
Level 4	Broadcast or CCI protected communication with station C	Any transmission	Q2d
NOTE 1: The decision criterion in this table is the distance between station A and station B. However, the requirement to check for CCI-protected communications at any given priority level requires station A to also examine the distance relationship between station B and station C, between station A and station C, between station B and station D and between station A and station D, as appropriate. It is possible to disable the selection process at any of the levels by setting the appropriate range constraint (Q2a to Q2d) to the largest possible value of 1 000 NM (see table 5.7). NOTE 2: For certain applications, Q2d could be set to zero (Q2d = 0) so that a slot can always be chosen even if this is at the expense of another application.			

Requirement reference	
5.2.4.3.2	In table 5.9, the following definitions and specifications shall apply: Station A — The station attempting to select a slot. Station B — A station that has previously reserved a slot. Station C — A station to which station A wishes to address a point-to-point communication. Station D — A station for which station B has reserved a slot for point-to-point communication. CCI protected — A point-to-point communication between two stations which fulfils the CCI conditions as defined in clause 5.2.3.3 and is therefore protected (its transmission can be heard by the intended recipient) if a third station simultaneously transmits in the same slot.

5.2.4.4 Parameter Q3 (replace queued data)

Requirement reference	
5.2.4.4.1	The parameter Q3 shall be a Boolean switch that shall be used to control queuing of repeated bursts on a congested channel.
5.2.4.4.2	If Q3 = TRUE, then a new data field shall replace a queued data field of the same type.
5.2.4.4.3	Otherwise, both the old and new data fields shall be transmitted.
NOTE: If a channel is busy and a synchronization burst containing ADS-B data cannot be transmitted, then a second synchronization burst (although with potentially different data) will overwrite the first burst.	

5.2.4.5 Parameter Q4 (number of available slots)

Requirement reference	
5.2.4.5.1	The parameter Q4 shall be used to control the number of slots added to the available slot list during the slot selection process (see clause 5.2.6.2).

5.2.5 Received transmission processing

Requirement reference	
5.2.5.1	Valid bursts shall be forwarded to the appropriate VSS user, along with the time of receipt of transmission.
5.2.5.2	The received signal quality and the time of receipt of the bursts shall be passed to the VME.
5.2.5.3	A station shall be capable of recognizing and processing all possible reservation types as defined in clauses 5.2.9 through 5.2.18.
5.2.5.4	When a station receives a burst with an unrecognized reservation type, it shall discard the burst without updating the reservation table.
5.2.5.5	When a station receives a known reservation type with an invalid subfield, or a known reservation type with valid subfields but an invalid combination, it shall reserve the slots indicated by the valid sub-fields, and not transmit a response, nor pass the burst to a VSS user.
5.2.5.6	When a station receives a burst with a known reservation type and a non-zero reserved subfield, it shall ignore the data in the reserved subfield.
5.2.5.7	The current slot for a burst shall be the slot in which the received transmission begins.
5.2.5.8	The burst length (bl) shall be the number of slots across which the burst is transmitted.
5.2.5.9	If the appropriate VSS user cannot be identified (for example, the message ID is reserved for future use, or that functionality is not implemented) and the burst contains one or more reservations for the receiving station only, then the station shall transmit a General Failure (see clause 5.2.20) with an error type of 00 hex or 80 hex (i.e. unsupported function) in the first slot of each of the reservations.
NOTE: Current slot and burst length (bl) are used throughout the text according to protocol definitions. In the text, unless otherwise stated, references to particular slot numbers (for example, for calculating the position of new reservations) are relative to the current slot which is taken to be slot 0. If a transmission extends across a slot boundary, it is considered to occupy the slots on both sides of the boundary for reservation purposes.	

5.2.6 Reserved access protocol specification

5.2.6.1 Reservation table

Requirement reference	
5.2.6.1.1	A station shall maintain a table of all reservations in the next 4 x M1 + 128 slots for all channels on which the station is transmitting and/or receiving.
5.2.6.1.2	For each reserved slot, the reservation table entry shall consist of the 27-bit address of the intended transmitter, the 27-bit address of the destination (if any) and the type of reservation made.
5.2.6.1.3	For periodic broadcast reservations (see clause 5.2.10) and directed request reservations (see clause 5.2.16), the reservation table shall also include pointers to all other reserved slots associated with the same reservation stream.
5.2.6.1.4	For the periodic broadcast protocol (see clause 5.2.10), the reservation table shall also record potential reservations, defined as the M1, 2 x M1, 3 x M1 and 4 x M1 slots after an occupied slot (see clause 5.1.5.3.1) for which no transmission has been decoded by the MAC layer.
5.2.6.1.5	For each potential reservation, the reservation table shall include the signal level (see clause 5.1.5.4) associated with the slot and the occupancy status as defined in clause 5.1.5.3.
5.2.6.1.6	Slots containing both potential reservations and reservations resulting from decoded transmissions shall be treated as if containing reservations from the decoded transmissions only.
5.2.6.1.7	The reservation table shall be updated before the end of the first slot after the end of the burst.
5.2.6.1.8	With the exception of transmissions using delayed bursts, and slots where a station has been directed to transmit by another station, a station shall wait for at least M1+ 128 slots after starting to listen to a channel before starting to transmit or reserve slots.
NOTE: Since a slot containing both a potential reservation and a reservation resulting from decoded transmission is treated as though only containing a decoded transmission, any potential reservations in subsequent superframes are effectively erased.	

5.2.6.2 Selecting slots for transmission or reservation

Requirement reference	
5.2.6.2.1	A station shall select slots for transmission or for reservation for later transmissions using the algorithm specified below.
5.2.6.2.2	The VSS user shall specify one or more groups of Quality of Service parameters Q2a, Q2b, Q2c, Q2d and Q4 for slot selection.
5.2.6.2.3	The station shall attempt to select slots using the first group of Quality of Service parameters.
5.2.6.2.4	If slot selection is unsuccessful, the station shall use the next group and continue with successive groups until a slot has been selected.
5.2.6.2.5	If, having used all groups of Quality of Service parameters, no slot has been selected, the VSS user shall be informed that slot selection has been unsuccessful.
	Specification of candidate slots
5.2.6.2.6	The VSS user shall specify one or more ranges of candidate slots for slot selection.
	Derivation of a list of available slots
	Slot selection criteria
5.2.6.2.7	A list of available slots shall be chosen from the candidate slots using the following rules:
5.2.6.2.8	All unreserved slots shall be added to the list of available slots (shown as level 0 in table 5.9).
5.2.6.2.8a	Slots that will be occupied by burst(s) that the station is currently preparing for transmission on any channel shall be considered as reserved slots for the purpose of slot selection.
5.2.6.2.9	If, having completed the action in clause 5.2.6.2.8, the number of available slots is less than Q4, further available slots shall be selected from slots that have been previously reserved by other stations.
5.2.6.2.10	The station shall initially select from slots which obey conditions specified as level 1 in table 5.9 until Q4 available slots have been chosen.
5.2.6.2.11	If, having applied level 1 conditions, the number of available slots is still less than Q4, slot selection shall continue using level 2 conditions.
5.2.6.2.12	The process shall continue using subsequent levels until Q4 slots have been selected or until all levels have been applied.
5.2.6.2.13	At each level, selection shall start with slots reserved by the most distant station and proceed in decreasing range order.
NOTE 1: The method for specifying candidate slots is protocol dependent (see clauses 5.2.9 to 5.2.18). NOTE 2: In addition to slots excluded because of ground quarantine, the VSS user can also specify other slots that should be excluded for the purposes of slot selection. Such slots might be slots that are potentially reserved (see clause 5.1.5.3) or which the VSS user does not wish to be used at all for slot selection. NOTE 3: In certain circumstances, for example when a channel is populated by a mixture of periodic broadcast and non-periodic broadcast reservations, it may be beneficial to additionally avoid slots containing potential reservations when selecting slots for protocols other than the periodic broadcast protocol (slot selection for which is covered by clauses 5.2.10.5.3 to 5.2.10.5.8).	

Requirement reference	
	Recommendation
5.2.6.2.14	In selecting the list of available slots at level 0, priority should be given to candidate slots which are not reserved for transmission on any channel monitored by the station, and which also do not violate quarantine constraints (see clause 5.2.3.2) on the desired transmit channel.
5.2.6.2.14aa	A mobile station should exclude from consideration any slots which have been previously reserved for a point-to-point transmission, on any channel, where it is the intended destination.
	Additional considerations for slot selection for transmission
5.2.6.2.14a	When selecting the list of available slots for transmission in a channel for mobile station A or for another mobile station B, station A shall exclude from consideration the specific slots which it knows are reserved for transmission for the intended station (either A or B) on other channels monitored by station A.
	Additional constraints applying to Global Signalling Channels (GSCs)
5.2.6.2.14b	On channels designated as GSCs (see Annex 10 - Aeronautical Telecommunications, Volume III, Part I Digital Data Communication System, chapter 6, 6.9.2.2.1 of ICAO VDL SARPs [2]), mobile stations maintaining primary or secondary time shall exclude the first V66 (see clause 5.2.17.5.4) slots of every UTC second. (See note).
5.2.6.2.14c	The first V66 slots after every UTC second shall comprise the Virtual Link Management Channel (VLMC) and shall be allocated for ground station use only.
	Selection of slots from available slots
5.2.6.2.15	If, having completed the derivation of a list of available slots, the number of available slots is zero, no slot shall be selected and the VSS user shall be informed that slot selection was unsuccessful.
5.2.6.2.16	If the number of available slots is greater than or equal to 1, a slot shall be chosen from the list of available slots such that the probability of choosing a given slot is the same as the probability of choosing any other slot.
	Selection of slots for burst lengths greater than 1
5.2.6.2.17	For burst lengths greater than 1, the process specified in clauses 5.2.6.2.7 to 5.2.6.2.14 shall be applied to continuous blocks of slots of length equal to the burst length.
5.2.6.2.18	A block of slots shall be regarded as available at a particular level number (see table 5.9) if all slots within the block are available at the same or lower level number.
5.2.6.2.19	The procedure described in clauses 5.2.6.2.15 to 5.2.6.2.16 shall then be used to select one of the available blocks.
	Limits on selection of reserved slots
5.2.6.2.20	A station which has selected a slot, that was reserved by another station shall not select another slot reserved by that station within M1 - 1 slots after the selected slot.
NOTE: The first V66 slots of every UTC second are allocated on a per frequency basis.	

5.2.6.3 Reserved transmissions

Requirement reference	
5.2.6.3.1	When a station has a burst to transmit for which it has a reservation, it shall transmit the scheduled data in the reserved slots, except as noted below.
	Unavailable data
5.2.6.3.2	If the data for a burst for which a slot was reserved by another station, is unavailable when it is time to transmit, then the station shall send a General Failure (see clause 5.2.20).
	Reservation no longer valid
5.2.6.3.3	A station shall check that a reservation is valid according to the procedures of clause 5.2.6.4 before transmitting.

5.2.6.4 Reservation conflicts

Requirement reference	
5.2.6.4.1	If a station, A, receives a burst containing a reservation from another station, B, for a slot which has already been reserved for station A to transmit, then station A shall take the following action:
5.2.6.4.2	If the conflicting reservation from station B also requires station A to transmit, then station A shall transmit (i) the response with the higher priority (as determined by Q1), or (ii) the first requested transmission in the case of equal priority (see note), or else;
5.2.6.4.3	If station A no longer requires to transmit in the existing reservation, or does not have the necessary information to transfer, then it shall not transmit in the slot, or else;
5.2.6.4.4	If the existing reservation for station A to transmit was made by a station other than A (i.e. by a unicast request (sdf = 0), information transfer, or directed request reservation), then A shall transmit in the slot in accordance with the existing reservation, or else;
5.2.6.4.5	If the existing reservation for station A to transmit was made by A itself, then A shall apply the procedure described in clauses 5.2.6.2.7 to 5.2.6.2.14 to determine whether, in the knowledge of the reservation made by station B, the slot is available at any level 1, 2, 3 or 4, using the same values of Q2 and other parameters as originally used to select the slot or other VSS user supplied QoS parameters for conflict resolution;
5.2.6.4.6	If the slot is determined to be available by this process, then A shall transmit according to its existing reservation;
5.2.6.4.7	If the slot is no longer available, the actions specified in table 5.10 shall be performed.
NOTE 1: If the priority of a burst is undefined, the default value for priority should be assumed. NOTE 2: The rules determine the action that a station takes in the event that a reservation conflict is detected. This is a normal event which is expected to occur as a result of slot reuse under CCI protection. In the event of a conflict, the slot selection criteria are generally reapplied to determine whether or not the slot could still have been selected in the knowledge of the new conflicting reservation. Generally, a station required to transmit in a slot that was reserved for it by another station will always transmit, since it cannot be assumed to have possession of the necessary information to determine the optimum action.	

5.2.6.5 Transmission conflicts for mobile stations

Requirement reference	
NOTE: The following rules describe the action that a mobile station will take when it detects a request for it to transmit simultaneously in the same slot on different channels.	
5.2.6.5.1	If a mobile station is requested to transmit in the same slot on different channels, then the station shall take the following action:
5.2.6.5.2	a) If there is only one transmission with the highest priority among the required transmissions, then the station shall transmit this highest priority transmission.
5.2.6.5.3	b) If there is only one ground-initiated transmission among the transmissions with the same highest priority, then the station shall transmit this ground-initiated transmission.
5.2.6.5.4	c) If there is more than one ground-initiated transmission among the transmissions with the same highest priority, then the station shall transmit the last requested of these ground-initiated transmissions.
5.2.6.5.5	d) If there are no ground-initiated transmissions among the transmissions with the same highest priority, then the station shall transmit the first requested transmission.

Table 5.10: Action in the event of reservation conflict

Protocol for A's existing reservation (made by A)	Protocol for B's conflicting reservation	Action by A
Slots reserved by station A using ground quarantine (see clause 5.2.3.2)	Any	Transmit according to existing reservation.
Periodic broadcast	Incremental broadcast, big negative dither unicast request, or information transfer	Transmit according to existing reservation.
Periodic broadcast	Periodic broadcast (autonomous/directed), directed request, slots reserved by ground quarantine (see clause 5.2.3.2)	If the conflict occurs later than A's next transmission in the stream, then select a new transmission slot and reduce the value of TV11 so as to cause the stream to dither to the new slot prior to the conflict; otherwise, set TV11 equal to 1 so that A's next transmission causes the stream to dither to a different slot in the next superframe after the superframe in which the conflict first occurs.
Incremental broadcast	Any	Do not transmit in the existing reservation, and make the transmission in an alternative slot by random access (see clause 5.2.7).

5.2.7 Random access protocol specification

5.2.7.1 General

Requirement reference	
5.2.7.1.1	The station shall implement a non-adaptive p-persistent algorithm to allow equitably all stations the opportunity to transmit while maximizing system throughput, minimizing transit delays, and minimizing collisions.
NOTE:	Transmissions which use the random access procedures may be used to place reservations for future transmissions that also use the reserved access procedures (clause 5.2.6) or they may be "one-off" transmissions which place no reservations and which will conform to either the null reservation burst format (clause 5.2.9) or the response protocol burst format (clause 5.2.18).

5.2.7.2 Random access parameters

Requirement reference	
5.2.7.2.1	Random access parameters shall be as described in table 5.11.

Table 5.11: Random access VSS system parameters

Symbol	Parameter name	Minimum	Maximum	Default	Increment
TM2	Channel busy timer	20 slots	4 500 slots	1 500 slots	20 slots
p	persistence	1 / 256	1	64 / 256	1 / 256
VS3	Maximum number of access attempts	1	65 536	24	1

Requirement reference	
	Timer TM2 (channel busy timer)
5.2.7.2.2	Timer TM2 indicates the number of slots (TM2) that a sublayer shall wait after receiving a request to transmit.
5.2.7.2.3	This timer shall be started if it is not already running, when the VSS sublayer receives a request for random transmission.
5.2.7.2.4	Upon a successful random transmission access attempt, the timer shall be cleared if the random transmit queue is empty and reset if it is not empty.
5.2.7.2.5	When the timer expires, the VSS user shall be informed that the channel is congested.
	Parameter p (persistence)
5.2.7.2.6	Parameter p shall be the probability that the station will transmit on any random access attempt.
5.2.7.2.7	If the station is able to select a slot, then the station shall transmit on the slot boundary with probability p.
	Counter VS3 (maximum number of access attempts)
5.2.7.2.8	Counter VS3 shall be used to limit the maximum number of random access attempts (VS3) that a station will make for any transmission request.
5.2.7.2.9	This counter shall be cleared upon system initialization, Timer TM2 expiring, or a successful access attempt.
5.2.7.2.10	The counter shall be incremented after every unsuccessful random access attempt.
5.2.7.2.11	When the counter reaches the maximum number of random access attempts, or when it has been cleared due to expiration of TM2, authorization to transmit shall be granted as soon as the channel becomes available.

5.2.7.3 Random access procedures

Requirement reference	
	Random access procedures
5.2.7.3.1	When the station has one or more bursts to transmit for which it does not have a reservation, it shall use a p-persistent algorithm as defined in ICAO VDL4 Technical Manual [1], with the additional constraints defined below:
5.2.7.3.2	Access attempts shall only be made and transmission shall only begin on a slot boundary of available slots.
5.2.7.3.3	A station shall regard a slot or block of slots as available for a random transmission if it conforms to the criteria of any of Levels 0 through 2 in table 5.9 using default or VSS user-supplied quality of service parameters.
5.2.7.3.4	Transmission shall not begin if the station has not previously made or received a reservation for the prior slot, and the slot is busy as defined in clause 5.1.5 at the slot boundary.
5.2.7.3.5	If the station is unable to select a slot, this shall be regarded as an unsuccessful random access attempt.
	Random access procedures for delayed transmissions
5.2.7.3.5a	Delayed transmissions shall use a p-persistent algorithm defined as follows:
5.2.7.3.5b	a) transmissions shall be delayed relative to the slot boundary in accordance with 5.1.6.3; and
5.2.7.3.5c	b) a station shall not start a transmission if the channel idle/busy status (see VDL SARPs clause 6.9.5.3) is busy at the intended (delayed) start time.
5.2.7.3.5d	If the station is unable to select a slot, this shall be regarded as an unsuccessful random access attempt.
5.2.7.3.5e	If the station is able to select a slot, then the station shall transmit on the slot boundary with probability p (as defined in clause 5.2.7.2.6).
	Recommendation
5.2.7.3.6	When possible, a station should use the reserved access protocols described in clause 5.2.6 to reserve slots for new transmissions by adding reservation fields to transmissions for which slots have already been reserved.
5.2.7.3.7	The random access protocol should be used only if there is no suitable opportunity to reserve a slot.
5.2.7.3.8	Void.
5.2.7.3.9	Void.
	Transmit queue management
5.2.7.3.10	There shall be a single queue for all random transmissions which do not have reserved slots for transmission.
5.2.7.3.11	This queue shall be sorted in priority order, with a higher value of Q1 being transmitted before a lower value of Q1.
5.2.7.3.12	If Q3 is TRUE, then the queue shall be searched to determine if a burst or frame of the same type has been queued, and the old data field replaced with the new data field if it has.
NOTE: Consider the case where a station intends to apply a p-persistent algorithm for random transmission at the start of slot k. If the prior slot [k−1] is reserved and slot k is unreserved or effectively unreserved, the station may be confident that the transmission in slot [k−1] will terminate and garble will not occur. However, if slot [k−1] is unreserved (according to the reservation table of the station) but nonetheless busy, the station has no way of knowing whether the transmission will terminate or continue. So in this case, a physical layer measurement is necessary to ensure that the transmission has terminated. Since the measurement process takes finite time, an apparently unreserved transmission which ends close to the end of slot [k−1] may forestall a random transmission in slot k. This is unavoidable.	

5.2.8 Void

5.2.9 Null reservation protocol specification

5.2.9.1 Null reservation burst format

Requirement reference	
5.2.9.1.1	A reservation ID (rid) = 1 and a reservation data (rd) field in accordance with table 5.12 shall indicate a null reservation.

Table 5.12: Null reservation bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
reservation data (rd) field	n - 3	x	x	x	x	x	x	0	0
	n - 2	0	0	0	0	0	0	0	0
NOTE: Bits denoted "x" are available for use within the information field (in).									

Requirement reference	
5.2.9.1.2	In this case, the information field shall extend up to the last 10 bits prior to the CRC.

5.2.10 Periodic broadcast protocol specification

5.2.10.1 Periodic broadcast reservation burst format

Requirement reference	
5.2.10.1.1	A reservation ID (rid) = 1 and a reservation field in accordance with table 5.13 shall indicate a periodic broadcast reservation.
5.2.10.1.2	In this case, the information field shall extend up to but excluding the last 10 bits prior to the CRC.
NOTE: The periodic broadcast protocol is intended for those VSS users which transmit one or more times per superframe for a number of superframes. A sequence of reserved slots linked by a periodic broadcast reservation is known as a "stream". A periodic broadcast reservation burst reserves a slot in the next superframe for its own stream (i.e. a VSS user transmitting 3 times a minute has 3 streams).	

Table 5.13: Periodic broadcast reservation bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
periodic timeout (pt)	n - 3	x	x	x	x	x	x	pt_2	pt_1
periodic offset (po)	n - 2	po_8	po_7	po_6	po_5	po_4	po_3	po_2	po_1
NOTE 1: Bits denoted "x" are available for use within the information field (in). NOTE 2: If the reservation field is all zeros, then a null reservation is being made (see clause 5.2.9). In the case of pt = 3, a combined periodic broadcast and incremental broadcast is indicated, in which case the periodic offset (po) subfield is replaced by the incremental offset (io) subfield, as described in clause 5.2.12. When io = 0 binary, only a periodic reservation is made.									

Requirement reference	
5.2.10.1.3	The subfields shall be as defined in table 5.14.

Table 5.14: Periodic broadcast reservation field encoding

Subfield	Range	Encoding
periodic offset (po)	-127 to +127	two's complement math po = -128 is invalid
periodic timeout (pt)	0 to 3	
NOTE: The two's complement form is one method of representing signed binary numbers with the aim to reduce the amount of binary states required to code a range of values that contain a shift in sign. In this case, it is –127 to +127. In two's complement, a positive number is represented as simple unsigned binary number, and negative number is represented as binary number that is added to a positive number of the same magnitude to get to zero. In practice this is done by changing all 0's to 1's and 1's to 0's and then by adding 1 to the result.		

Requirement reference	
5.2.10.1.4	po shall identify a slot relative to the first slot of the transmission in a future superframe.
5.2.10.1.5	pt shall define the number of superframes in the future for which a reservation is being made.

5.2.10.2 Periodic broadcast timers

Requirement reference	
	Timer TV11 (reservation hold timer)
5.2.10.2.1	The timer TV11 shall control the number of successive superframes which will use the same slot for transmission (see clause 5.2.10.5) before moving to a new slot.
5.2.10.2.2	There shall be one TV11 timer for each slot used for periodic broadcasts.

5.2.10.3 Periodic broadcast parameters

Requirement reference	
5.2.10.3.1	The periodic broadcast protocol shall implement the system parameters defined in table 5.15.

Table 5.15: Periodic broadcast VSS system parameters

Symbol	Parameter name	Minimum	Maximum	Recommended default	Increment
TV11min	Reservation hold timer minimum value	0 superframes	15 superframes	4 superframes	1 superframe
TV11max	Reservation hold timer maximum value	1 superframe	16 superframes	8 superframes	1 superframe
V11	Nominal periodic rate	1 per superframe	60 per superframe	1 per superframe	1 per superframe
V12	Periodic dither range	(2/M1) x V11	{truncate[M1/(2 x V11)]}(2/M1) x V11	{truncate[M1/(20 x V11)]}(2/M1) x V11	(2/M1) x V11

Requirement reference	
5.2.10.3.2	TV11 min shall be less than or equal to TV11 max.
5.2.10.3.3	The VSS user shall provide any of the parameters TV11 min, TV11 max, V11, V12 and Quality of Service parameters (Q2a to Q2d and Q4) for which the default values are not desired.
	Parameters TV11min and TV11max (reservation hold timer minimum and maximum values)
5.2.10.3.4	Parameters TV11min and TV11max shall be used to determine the start value for the TV11 timer, consistent with the procedure defined in clause 5.2.10.5.14.
	Parameter V11 (nominal periodic rate)
5.2.10.3.5	The parameter V11 shall be the number of times per superframe that a VSS user will transmit a burst.
	Parameter V12 (periodic dither range)
5.2.10.3.6	The parameter V12 shall define the range for candidate slots on either side of the nominal slot (see clauses 5.2.10.5.1 to 5.2.10.5.2) from which the station shall choose a slot or group of slots to be reserved for transmission once the TV11 timer expires.
5.2.10.3.7	V12 shall be specified as a fraction of the nominal periodic rate.
NOTE: The selected slot may be chosen from a range between the nominal slot {− (V12/2) x (M1/V11)} and the nominal slot {+ (V12/2) x (M1/V11)}. If this range is greater than ± 127, then the selected slot may be chosen from a range between nominal slot −127 and the nominal slot +127.	

5.2.10.4 Periodic broadcast reception procedures

Requirement reference	
5.2.10.4.1	Upon receipt of a burst containing a periodic broadcast reservation, the station shall update its reservation table and carry out the actions as specified in table 5.16.

Table 5.16: Action on receipt of periodic broadcast reservation burst

Periodic offset (po)	Periodic timeout (pt)	Action
0	0	No reservation (see note 1).
Any except 0	0, 1 or 2	Reserve the following slots for the source to broadcast: if pt = 1 or 2 then for j = 1 to pt, the slots equal to (j x M1) through (bl - 1 + (j x M1)) after the first slot of the received burst AND for j = pt + 1 to 4, the slots equal to (po + (j x M1)) through (bl - 1 + (po + (j x M1))) slots after the first slot of the received burst.
0	1 or 2	Reserve the following slots for the source to broadcast: for j = 1 to pt, the slots equal to (j x M1) through (bl - 1 + (j x M1)) after the first slot of the received burst.
any	3	Reserve the following slots for the source to broadcast: for j = 1 to 4, the slots equal to (j x M1) through (bl - 1 + (j x M1)) after the first slot of the received burst (see note 2).
NOTE 1: Reservation format is the same as null reservation (see clause 5.2.9). NOTE 2: The interpretation of the periodic offset subfield in the case of periodic timeout = 3 and io <> 0 binary is described in clause 5.2.12.		

Requirement reference	
5.2.10.4.2	All reservations associated with a single periodic broadcast reservation burst shall be known as a stream.
5.2.10.4.3	The actions defined in table 5.16 shall cancel any previous reservations for the same stream.
5.2.10.4.4	If a station was expecting to receive a transmission from a peer station containing a periodic broadcast reservation, but receives a transmission from the peer station containing an incremental reservation (see clause 5.2.11) or a unicast request with the source/destination flag set equal to 1 (see clause 5.2.14), the station shall cancel the periodic broadcast reservation stream for the peer station.

5.2.10.5 Periodic broadcast transmission procedures

Requirement reference	
	Selection of nominal slots
5.2.10.5.1	When periodic broadcast transmissions are used on a frequency which is not subject to directed-slot reservations (see clauses 5.2.16.1.7 to 5.2.16.1.9), a station shall select nominal slots (n_slot) which form a periodic sequence in time, with a variation of no more than ±1 slot as required to accommodate the constraints imposed by the nominal reporting rate for the application, such that the nominal slot for the kth stream occurs truncate[(k - 1) x (M1/V11)] ± 1 after the nominal slot for the first stream and k = 2 to V11.
	Recommendation
5.2.10.5.2	When operating with a mixture of directed-slot reservations, autonomous, and/or directed-rate reservations (see clauses 5.2.16.1.7 to 5.2.16.1.9) for a given VSS User application which requires periodic broadcast transmissions, a station should aim to select nominal slots (n_slot) for the autonomous or directed rate which form a periodic sequence in time, considering all frequencies used.
5.2.10.5.2a	A station should shift all the nominal slots (n_slots) associated with an application's autonomous or directed rate reservations forward or backward in time without changing their relative positions as defined by clause 5.2.10.5.1, in order to enhance the likelihood of finding appropriate transmission slots for the application as a whole.
NOTE 1: For an application that requires periodic broadcast transmissions on multiple frequencies and for which no directed slot reservations have been received, the aggregate of all required transmissions should be used when calculating the nominal update rate (nr). Example 1: Two frequencies have a required nr of once per 10 seconds on each frequency. In this case, the nominal slots (n_slots) should be interleaved and equally spaced to achieve an aggregate nr of once per 5 seconds (i.e. considering the two channels together). Example 2: Two frequencies have a required nr of once per 15 seconds on frequency F1 and once per 5 seconds on frequency F2. In this case, the aggregate nr should be once per 3,75 seconds, with three successive n_slots on F2 spaced 3,75 seconds apart, followed by a 7,5 second gap centred on a n_slot for F1, followed by another three successive n_slots on F2 etc. NOTE 2: A station may shift individual slots or sets of slots as required in order to satisfy the needs in the selection of nominal slots (n_slots). This may be required, for example, if the application adds a new frequency or if the nominal update rate (nr) on one of the existing frequencies is changed in real time (i.e. with a directed rate request). NOTE 3: Clause 5.2.10.5.1 is relevant for transmissions using periodic reservations. It does not apply to transmissions using other reservation types or random transmissions. For example, transmissions made using random and incremental protocols are excluded.	

Requirement reference	
	Selection of slots for a periodic broadcast transmission
5.2.10.5.3	If there is no existing periodic reservation for the VSS user, the station shall select a current transmission slot (ct_slot) corresponding to each nominal slot by inspection of the reservation table data, using the following procedure:
5.2.10.5.4	The station shall use the slot selection procedure specified in clause 5.2.6.2 using all slots that are within (V12/2) × (M1/V11) of n_slot and within 127 slots of n_slot, as candidate slots, and the default or other VSS user supplied quality of service parameters.
5.2.10.5.5	When applying the slot selection procedure specified in clause 5.2.6.2, the station shall first select available slots at levels 0,1 and 2, excluding slots containing potential reservations associated with occupied slots as defined in clause 5.1.5.3.
5.2.10.5.6	Selections at level 0 shall select from slots containing potential reservations associated with unoccupied slots in increasing order of signal level as defined in clause 5.1.5.4.
5.2.10.5.7	If, on completion of the selection of available slots at level 2, less than Q4 slots have been chosen, the station shall select from slots containing potential reservations associated with occupied slots in increasing order of signal level as defined in clause 5.1.5.4.
5.2.10.5.8	If at the end of this process, less than Q4 slots have been chosen, the station shall then continue the slot selection process at level 3.
NOTE: The station tries to find unreserved slots in the range of [(V12 • M1)/V11] on either side of the nominal slot (n_slot) ignoring slots that are four superframes after a slot in which a station detects the presence of a transmission but is unable to decode the transmission. The ignored slots are assumed to contain potential reservations since the undecoded transmission is most likely to contain a periodic reservation protocol for subsequent superframes. If slot selection is unsuccessful by the end of level two, the potential reservations are then added back into the candidate range and selected in order of increasing signal level.	

Requirement reference	
	Calculation of slot availability
5.2.10.5.9	After selection of a new current transmission slot, the station shall compute the slot availability (s_avail), indicating how many consecutive superframes are available until the equivalent slot is reserved by another user.
5.2.10.5.10	The value of s_avail shall indicate the slot (ct_slot + s_avail × M1) which is reserved by another user, and range from 1 (for a slot that is reserved in the next superframe) to 4 (for slots that currently have no reservation for at least 3 superframes).
5.2.10.5.11	The calculation of s_avail shall use the following rules:
5.2.10.5.12	If the current transmission slot has not been previously reserved, s_avail shall be the number of superframes that are left before the equivalent slot is reserved;
5.2.10.5.13	If the current transmission slot has been previously reserved by a station, s_avail shall be the number of superframes that are left before the equivalent slot is reserved by a different user.
	Transmission in a new slot
5.2.10.5.14	If there is no prior reservation or if the station is using for the first time a slot for which there has been a prior reservation, the station shall start the timer TV11 at a value equal to s_avail, if s_avail = 1, 2 or 3, and otherwise equal to a random value uniformly chosen between TV11 min and TV11 max.
	Transmission for TV11 greater than 3
5.2.10.5.15	If the TV11 timer is greater than 3 and there is no requirement to associate the current transmission with an incremental reservation, the station shall transmit a burst containing a periodic broadcast reservation in the current transmission slot with io = 0 and pt = 3.
5.2.10.5.16	After transmission, the timer TV11 shall be decremented by one and the current transmission slot shall be incremented by M1.
	Reservation of a new slot for TV11 equal to 1, 2, or 3
5.2.10.5.17	If the TV11 timer is equal to 1, 2 or 3 and if the VSS user requires that periodic broadcast reservations are maintained after the current transmission slot reservation expires, the station shall reserve a future transmission slot (ft_slot) for subsequent transmissions.
5.2.10.5.18	If a future transmission slot has already been selected, there shall be no further slot selection.
5.2.10.5.19	Otherwise, selection of ft_slot shall be carried out using the procedure set out in clauses 5.2.10.5.3 to 5.2.10.5.8 using all slots that are within (V12/2) × (M1/V11) of n_slot and within 127 slots of n_slot and within 127 slots of ct_slot, except slot (ct_slot + TV11 × M1), as candidate slots.
	Transmission for TV11 equal to 1, 2 or 3
5.2.10.5.20	If the TV11 timer is equal to 1, 2 or 3 the station shall transmit a burst containing a periodic broadcast reservation in the current transmission slot with po = (ft_slot - ct_slot) and pt = TV11 - 1.
5.2.10.5.21	If a future transmission slot has not been selected and the VSS user does not require the reservation to be maintained, the value of po shall be set to zero.
5.2.10.5.22	After transmission, the timer TV11 shall be decremented and the current transmission slot set equal to ct_slot + M1.
	TV11 equal to zero
5.2.10.5.23	If the TV11 timer is equal to zero, and the VSS user requires a reservation to be maintained, then if a new slot has not been selected for further periodic broadcasts, the station shall select a new current transmission slot using the procedures set out in clauses 5.2.10.5.3 to 5.2.10.5.8.
5.2.10.5.24	If a new slot has been selected for further periodic broadcasts, the station shall set the current transmission slot equal to the future transmission slot.
5.2.10.5.25	The station shall start to transmit in the new current transmission slot carrying out the procedures set out in clauses 5.2.10.5.9 to 5.2.10.5.24.
5.2.10.5.26	If the VSS user does not require a reservation to be maintained, no further action shall be taken.
	Reservation cancellation
5.2.10.5.27	A station wishing to cancel a stream or reservations for its own transmissions, in the absence of a reservation conflict, shall transmit a periodic broadcast reservation burst with po = 0 and pt = 0 in the next reserved slot and the timer TV11 shall be cleared.
5.2.10.5.28	A station receiving such a burst shall clear all reservations known to be associated with the stream.

	Reservation cancellation
5.2.10.5.29	A station with less than two available slots within the dither range of any periodic stream shall move that stream and move any other streams necessary (see clause 5.2.10.5.2a).
NOTE 1: The reservation for a new slot is maintained for TV11 superframes unless slot availability (s_avail) indicates that only 1, 2 or 3 superframes are available before another station has placed a reservation.	
NOTE 2: The process in clause 5.2.10.5.17 selects a new slot to which the periodic broadcast transmission will move in the TV11 superframes after the current transmission slot (ct_slot). This new slot will occupy a different position in the superframe to the ct_slot.	
NOTE 3: Because all existing reservations for a stream are cancelled on receipt of a periodic reservation burst and are replaced according to the reservation information contained in the burst (see clause 5.2.10.4), this action has the effect of cancelling the whole stream.	
NOTE 4: Reservation conflicts are mediated in accordance with clause 5.2.6.5, and the requirement to transmit a reservation cancellation (i.e. po = 0 and pt = 0) does not apply if the transmission would be disallowed by the considerations outlined in clause 5.2.6.5.	

5.2.11 Incremental broadcast protocol specification

5.2.11.1 Incremental broadcast reservation burst format

Requirement reference	
5.2.11.1.1	A reservation ID (rid) = 0 with extended reservation ID and reservation fields set in accordance with table 5.17 shall indicate an incremental broadcast reservation.

Table 5.17: Incremental broadcast reservation bit encoding

Description	Octet	Bit Number							
		8	7	6	5	4	3	2	1
	n - 3	x	x	x	x	x	x	io_8	io_7
incremental offset (io)	n - 2	1	0	io_6	io_5	io_4	io_3	io_2	io_1
NOTE: Bits denoted "x" are available for use within the information field.									

Requirement reference	
5.2.11.1.2	In this case, the information field shall extend up to but excluding the last 10 bits prior to the CRC.
5.2.11.1.3	The subfields shall be as defined in table 5.18.

Table 5.18: Incremental broadcast reservation field encoding

Subfield	Range	Encoding
incremental offset (io)	0 to 255	(see clause 5.2.11.4)

Requirement reference	
5.2.11.1.4	io shall identify a slot relative to the first slot of the transmission.

5.2.11.2 Incremental broadcast parameters

Requirement reference	
5.2.11.2.1	The incremental broadcast protocol shall implement the system parameters defined in table 5.19.

Table 5.19: Incremental broadcast VSS system parameters

Symbol	Parameter name	Minimum	Maximum	Recommended default	Increment
V21	Nominal incremental period (slots)	16	1008	75	1
V22	Maximum incremental dither range (slots)	4	truncate[MIN(V21/2,125)]	max{truncate[MIN(V21/4, 62)], 8}	1
NOTE: The maximum value of V21 is set by the maximum allowed value of the incremental offset (io) subfield. The minimum value of V21 and V22 is set to ensure that there are at least 2 candidate slots from which to choose a slot to be reserved.					

Requirement reference	
5.2.11.2.2	The VSS user shall provide any of the parameters V21, V22 and Quality of Service parameters (Q2a to Q2d and Q4) for which the default values are not desired.
	Parameter V21 (nominal incremental period)
5.2.11.2.3	The parameter V21 shall be the nominal time after the first slot of the incremental broadcast transmission that a VSS user will transmit a burst.
	Parameter V22 (maximum incremental dither range)
5.2.11.2.4	The parameter V22 shall define the range for candidate slots on either side of the nominal slot from which the station shall choose a slot or group of slots to be reserved for transmission.
5.2.11.2.5	V22 shall be specified as an integer number of slots.
NOTE: The selected slot may be chosen from a time range between [V21 ± (V22 × V21)].	

5.2.11.3 Incremental broadcast reception procedures

Requirement reference	
5.2.11.3.1	Upon receipt of a burst containing an incremental broadcast reservation, a station shall reserve the slot equal to (4 × io) through (bl - 1 + 4 × io) after the first slot of the received burst for the source to broadcast.
5.2.11.3.2	When a burst contains an incremental broadcast reservation with io = 0, then no incremental reservation shall be placed.

5.2.11.4 Incremental broadcast transmission procedures

Requirement reference	
	Selection of the transmission slot for the incremental broadcast reservation
5.2.11.4.1	If no slot or group of consecutive slots, has been reserved for transmission of an incremental reservation, and if the incremental reservation is not to be combined with a periodic broadcast reservation (see clause 5.2.12), the station shall select a slot or group of consecutive slots using the random access procedures (see clause 5.2.7).
5.2.11.4.2	The transmission slot (t_slot) shall be the first slot of the incremental broadcast transmission.
	Selection of the reserved slot for the incremental broadcast reservation
5.2.11.4.3	The station shall choose a slot or group of consecutive slots to reserve using the slot selection procedure specified in clause 5.2.6.2: a) using VSS user supplied quality of service parameters; and b) candidate slots in the range V21 - V22 through V21 + V22 such that the chosen slot, or the first slot in the chosen group of slots, is an exact modulo 4 difference from t_slot.
5.2.11.4.4	The reserved slot (r_slot) shall be the chosen slot or the first slot in the chosen group of slots.
	Incremental broadcast burst transmission
5.2.11.4.5	The station shall transmit an incremental broadcast burst in the transmission slot with the value of io set to (r_slot - t_slot) / 4, and update the value of t_slot for each transmission.

5.2.12 Combined periodic broadcast and incremental broadcast protocol specification

5.2.12.1 Combined periodic broadcast and incremental broadcast reservation burst

Requirement reference	
5.2.12.1.1	A reservation ID (rid) = 1 and a reservation field in accordance with table 5.20 shall indicate a combined periodic broadcast and incremental broadcast reservation.

Table 5.20: Combined periodic/incremental broadcast reservation bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
periodic timeout (pt) = 3	n - 3	x	x	x	x	x	x	1	1
incremental offset (io)	n - 2	io_8	io_7	io_6	io_5	io_4	io_3	io_2	io_1
NOTE: Bits denoted x are available for use within the information field.									

Requirement reference	
5.2.12.1.2	In this case, the information field shall extend up to the last 10 bits prior to the CRC.
5.2.12.1.3	The periodic timeout (pt) subfield shall be set to 3.
5.2.12.1.4	The incremental offset subfield (io) shall be as defined in clause 5.2.11.1.
5.2.12.1.5	All other parameters and procedures shall be as specified in clauses 5.2.10 and 5.2.11.

5.2.13 Big negative dither (BND) broadcast protocol specifications

5.2.13.1 BND reservation burst format

Requirement reference	
5.2.13.1.1	A reservation ID (rid) = 0, an extended reservation ID (erid) = 00001 binary, and reservation data set in accordance with table 5.21 shall indicate a Big Negative Dither (BND).

Table 5.21: BND broadcast reservation bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
negative dither (nd)	n - 3	x	x	x	x	x	x	nd_5	nd_4
extended reservation ID (erid)	n - 2	0	0	0	0	1	nd_3	nd_2	nd_1
NOTE: Bits denoted x are not used by this reservation type and shall be available for use within the information field.									

Requirement reference	
5.2.13.1.2	The subfields shall be as defined in table 5.22.

Table 5.22: BND broadcast reservation parameters

Subfield	Range	Encoding	Definitions
negative dither (nd)	0 to 31	See clause 5.2.13.3	nd identifies a slot relative to and earlier than the current slot + M1 - 128 slots.

5.2.13.2 BND broadcast parameters

Requirement reference	
5.2.13.2.1	There are no BND parameters.

5.2.13.3 BND broadcast reception procedures

Requirement reference	
5.2.13.3.1	Upon receipt of a burst containing a BND broadcast reservation, a station shall reserve the slots from (M1 - 128 - (4 × nd)) through (M1 - 128 - (4 × nd) + (bl - 1)) after the first slot of the received burst for the source to broadcast.
NOTE: This reservation type allows a station to place a reservation for a future transmission in a slot which is likely to be unreserved (see associated BDN broadcast transmission procedures in clause 5.2.13.4).	

5.2.13.4 BND broadcast transmission procedures

Requirement reference	
5.2.13.4.1	A station shall not transmit a BND reservation until it has listened to a channel for at least 254 slots and can select a slot for transmission from among those slots addressable by the BND reservation that uses the slot selection procedures defined in clause 5.2.6.2 and the VSS user-supplied QoS parameters, namely Q2 and Q4.

5.2.14 Unicast request protocol specification

5.2.14.1 Unicast request reservation burst format

Requirement reference	
5.2.14.1.1	A reservation ID (rid) = 0 with an extended reservation ID and reservation fields set in accordance with table 5.23 shall indicate a unicast request reservation.
NOTE: This protocol is intended for a VSS user which requires (a) a response from a peer VSS user in a reserved slot, (b) a slot to be reserved for a transmission to a peer, or (c) a slot to be reserved to make a broadcast transmission. In the case of (c), the protocol is a more flexible version of the incremental broadcast protocol, supporting reservations of variable length on user defined channels.	

Table 5.23: Unicast request reservation bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
destination address (d)	n - 8	d_{24}	d_{23}	d_{22}	d_{21}	d_{20}	d_{19}	d_{18}	d_{17}
	n - 7	d_{16}	d_{15}	d_{14}	d_{13}	d_{12}	d_{11}	d_{10}	d_9
	n - 6	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1
source/destination flag (sdf)	n - 5	ro_{12}	ro_{11}	ro_{10}	ro_9	sdf	d_{27}	d_{26}	d_{25}
response offset (ro)	n - 4	ro_8	ro_7	ro_6	ro_5	ro_4	ro_3	ro_2	ro_1
length (lg)	n - 3	res	res	res	res	lg_4	lg_3	lg_2	lg_1
priority (pr)	n - 2	0	0	1	0	pr_4	pr_3	pr_2	pr_1
NOTE: "res" refers to bits available for the information field.									

Requirement reference	
5.2.14.1.2	The subfields and associated actions shall be as defined in table 5.24.

Table 5.24: Unicast request reservation field encoding

Subfield	Range	Encoding/Actions	Definitions
response offset (ro)	0 to 4 095		ro identifies a slot relative to the first slot of the transmission.
destination address (d)	0 to 2^{27}-1	See clause 5.3.1.3.	d is the 27-bit address of the destination station.
source/destination flag (sdf)	Boolean	If sdf = 0, reserve the response slot for the destination station to transmit. If sdf = 1, reserve the response slot for the source station to transmit.	sdf indicates which station will respond in the reserved response slot. Note that the source station is the station placing the reservation.
length (lg)	0 to 15		lg is one less than the number of slots that are reserved for the response.
priority (pr)	0 to 15	See table 5.8.	

Requirement reference	
5.2.14.1.3	In the case that the address type field (see clause 5.3.1.2.1) is equal to 7, bits 1 through 24 of the destination subfield (d) shall be absent, so that the information field will extend up to the last four octets prior to the CRC.
5.2.14.1.4	Otherwise, the burst shall include all of the destination subfield (d), so that the information field will extend up to the last seven octets prior to the CRC.

5.2.14.1a Unicast request parameters

Requirement reference	
5.2.14.1a.1	The unicast request protocol shall implement the system parameters as defined in table 5.24a.

Table 5.24a: Unicast request VSS system parameters

Symbol	Parameter name	Minimum	Maximum	Recommended default	Increment
V32	Minimum response delay	2 slot	500 slots	20 slots	1 slot
V33	Maximum response delay	2 slot	4 095 slots	1 000 slots	1 slot
V34	Source/destination control	0	1	0	1
V35	Broadcast control	0	1	0	1
V36	Length of reserved block	1 slot	256 slots	N/A	1 slot

Requirement reference	
5.2.14.1a.2	The VSS user shall provide the destination address and any of the parameters V32, V33, V34, V35, V36 and Quality of Service parameters (Q2a to Q2d, Q4 and Q5) for which the default values are not desired.
	Parameter V32 (minimum response delay)
5.2.14.1a.3	Parameter V32 shall be the minimum delay, measured in slot intervals, that a station will provide to a responder in order to ensure that the responder can generate the response before its reserved slot (see note 1).
	Parameter V33 (maximum response delay)
5.2.14.1a.4	Parameter V33 shall be the maximum delay, measured in slot intervals, that a station will provide to a responder in order to ensure timely delivery in case a retransmission is required.
	Parameter V34 (source/destination control)
5.2.14.1a.5	Parameter V34 shall control whether the unicast reservation protocol is used to reserve a slot for the destination station to transmit a response to the source (V34 = 0) or for the source station to transmit a response to the destination (V34 = 1).
5.2.14.1a.6	If the broadcast control parameter (V35 = 1), the value of V34 shall be ignored (see note 2).
	Parameter V35 (broadcast control)
5.2.14.1a.7	Parameter V35 shall control whether the lowest 24 bits of the destination subfield (d) are included in the reservation.
5.2.14.1a.8	If V35 = 0, then the lowest 24 bits of the destination subfield shall be included and the reservation will be for the station to transmit to or receive from a peer station.
5.2.14.1a.9	Otherwise the lowest 24 bits of the destination subfield shall be omitted, the address type field shall be set to 7 and the reservation will be for the station to make a broadcast transmission.
	Parameter V36 (length of reserved block)
5.2.14.1a.10	Parameter V36 shall be the number of reserved slots required for the unicast reservation protocol response.
NOTE 1: V32 × 60/M1 is the maximum time that a station has to generate a response to the request. NOTE 2: If the destination subfield is omitted (V35 = 1), then the reservation is for the source to broadcast and the value of V34 has no meaning.	

5.2.14.2 Unicast request reception procedures

Requirement reference	
5.2.14.2.1	Upon receipt of a burst containing a unicast request reservation, a station shall reserve all of the slots from (1 + ro) through (1 + ro + lg) after the first slot of the received burst for: a) the destination to transmit a response to the source (if sdf = 0 and address type field <> 7); b) or for the source to transmit a response to the destination (if sdf = 1 and address type field <> 7); c) or for the source to make a broadcast transmission (if address type field = 7).

5.2.14.3 Unicast request transmission procedures

Requirement reference	
	Selection of the transmission slot for the unicast request reservation
5.2.14.3.1	If no slot has been reserved for transmission of a unicast reservation, the station shall select a slot using the random access procedures (see clause 5.2.7).
5.2.14.3.2	The transmission slot (t_slot) shall be the slot containing the unicast request reservation transmission.
	Selection of the reserved slot for the response
5.2.14.3.3	A block of slots of length V36 to be reserved for the response (address type field ≠ 7) or broadcast transmission (address type field = 7) shall be selected using the slot selection procedure specified in clause 5.2.6.2, using VSS user supplied quality of service parameters, and candidate slots in the range V32 to V33 after the transmitted burst.
5.2.14.3.4	The reserved slot (r_slot) shall be the chosen slot or the first slot in the chosen group of slots.
	Unicast request burst transmission
5.2.14.3.5	A station sending a unicast request burst to its peer (V35 = 0) shall include the unicast request reservation field.
5.2.14.3.6	It shall set the destination (d) subfield to the destination of the burst, the response offset (ro) subfield to a value of (r_slot - t_slot -1), the length (lg) subfield equal to (V36 - 1), the priority (pr) subfield equal to the priority of the burst to be transmitted as defined by Q1 and the source/destination flag (sdf) to V34.
5.2.14.3.7	A station sending a unicast request burst to reserve a slot for a subsequent broadcast (V35 = 1) shall include the unicast request reservation field.
5.2.14.3.8	It shall set the response offset (ro) subfield to a value of (r_slot - t_slot - 1), the length (lg) subfield equal to (V36 - 1), the priority (pr) subfield equal to the priority of the burst to be transmitted as defined by Q1 and the address type field equal to 7.
	Retransmission after no response
5.2.14.3.9	In the case of address type subfield ≠ 7 and sdf = 0, if a response is not received by the end of the reserved response slot(s), then the station shall retransmit the unicast burst according to the procedures of clause 5.2.21.
	Slot selection criteria for unicast request with sdf = 1
5.2.14.3.10	A station applying the slot selection criteria of clause 5.2.6.2.7 to 5.2.6.2.13 shall exclude any slot reserved by another station using the unicast request protocol with sdf = 1.
NOTE: The use of the unicast request protocol with sdf = 1 allows a station placing the reservation to transmit to a different destination than the destination included in the reservation field. This ability allows a station completing a data transfer with one station to simultaneously start a new data transfer to a different destination. However, since the new destination address is not known by any other station, it is not possible to apply CCI criteria and the slot must be excluded from slot selection.	

5.2.15 Information transfer request protocol specification

5.2.15.1 Information transfer request reservation burst format

Requirement reference	
5.2.15.1.1	A reservation ID (rid) = 0 with extended reservation ID (erid) = 01010 binary and reservation fields set in accordance with table 5.25 shall indicate an information transfer request reservation.
NOTE: This protocol is intended for a VSS user which requires a peer VSS user to send a response of length (lg). The protocol also allows the requesting VSS user to place a reservation for an acknowledgement by the requesting VSS user to the response field.	

Table 5.25: Information transfer request reservation bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
acknowledgement offset (ao)	n - 10	res	ao_7	ao_6	ao_5	ao_4	ao_3	ao_2	ao_1
length (lg)	n - 9	res	res	res	res	lg_4	lg_3	lg_2	lg_1
response offset (ro)	n - 8	ro_8	ro_7	ro_6	ro_5	ro_4	ro_3	ro_2	ro_1
	n - 7	ro_{12}	ro_{11}	ro_{10}	ro_9	f_{12}	f_{11}	f_{10}	f_9
frequency (f) destination address (d)	n - 6	f_8	f_7	f_6	f_5	f_4	f_3	f_2	f_1
	n - 5	d_{24}	d_{23}	d_{22}	d_{21}	d_{20}	d_{19}	d_{18}	d_{17}
	n - 4	d_{16}	d_{15}	d_{14}	d_{13}	d_{12}	d_{11}	d_{10}	d_9
	n - 3	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1
	n - 2	0	1	0	1	0	d_{27}	d_{26}	d_{25}
NOTE: "res" refers to bits available for the information field.									

Requirement reference	
5.2.15.1.2	In this case, the information field shall extend up to the last nine octets prior to the CRC.
5.2.15.1.3	The subfields shall be as defined in table 5.26.

Table 5.26: Information transfer reservation field encoding

Subfield	Range	Encoding
length (lg)	See table 5.24	lg is one less than the number of slots that are reserved for the response.
acknowledgement offset (ao)	0 to 127	ao identifies a slot relative to the end of the block of slots reserved by the response offset and length subfields.
response offset (ro)	See table 5.24	ro identifies a slot relative to the first slot of the transmission.
destination address (d)	See clause 5.3.1.3	d is the 27-bit address of the destination station for which the block of slots is being reserved.
frequency (f)	bit 12: frequency band indicator: 0: VHF band 108 MHz to 137,000 MHz 1: reserved for future allocation bits 1 to 11: frequency allocation for bit 12 = 0: 1 to 1 160 per frequency band in 25 kHz increments. 1 161 to 2 047 reserved for future allocation. 1 indicates bottom of band. f = 001 hex = 108,000 MHz f = 000 hex indicates the frequency on which the burst is transmitted.	The frequency subfield (f) identifies the frequency on which the reservation is to be made for the response.

5.2.15.2 Information transfer request reception procedures

Requirement reference	
5.2.15.2.1	Upon receipt of a burst containing an information transfer request reservation, a station shall reserve on the specified frequency all of the slots from (1 + ro) through (1 + ro + lg) after the first slot of the received burst for the destination to transmit one or more information bursts to the source.
5.2.15.2.2	Also, the slot equal to (2 + ro + lg + ao) after the first slot of the received burst shall be reserved for the source to transmit an acknowledgement to the destination on the same frequency as the burst containing the information transfer request reservation.

5.2.16 Directed request protocol specification

5.2.16.1 Directed request reservation burst format

Requirement reference	
5.2.16.1.1	A reservation ID (rid) = 0, an extended reservation ID (erid) = 01100 binary, and reservation fields set in accordance with table 5.27 shall indicate a directed request reservation.
NOTE 1: This protocol is intended for a VSS user which is responding to a plea for slot reservations (rapid network entry), or which requires periodic broadcast responses from a peer VSS user. Both of these scenarios involve reservations calculated and declared for use by the peer station. In addition, this protocol allows a VSS user to request that a peer VSS user autonomously transmit at a specified rate. NOTE 2: The directed request reservation burst may be used for a plea response, autotune, or other directed request application.	

Table 5.27: Directed request reservation bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
identification of additional reservation data	n - 15								
	n - 14								
	n - 13								
	n - 12								
	n - 11		per table 5.29, table 5.31 through table 5.32						
	n - 10								
	n - 9								
	n - 8								
	n - 7								
nominal update rate (nr); plea response flag (pr_flag)	n - 6				pr_flag	nr_4	nr_3	nr_2	nr_1
destination address (d)	n - 5	d_{24}	d_{23}	d_{22}	d_{21}	d_{20}	d_{19}	d_{18}	d_{17}
	n - 4	d_{16}	d_{15}	d_{14}	d_{13}	d_{12}	d_{11}	d_{10}	d_9
	n - 3	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1
extended reservation ID (erid)	n - 2	0	1	1	0	0	d_{27}	d_{26}	d_{25}

Requirement reference	
5.2.16.1.2	The length of the reservation field shall be determined by the value of the plea response flag (pr_flag).
5.2.16.1.3	For the case of pr_flag = 1, the information field shall extend up to the last fourteen octets prior to the CRC.
5.2.16.1.4	For the case of pr_flag = 0, the information field shall extend up to the last ten octets prior to the CRC.
5.2.16.1.5	The nominal update rate (nr) field shall be encoded in accordance with table 5.28.

Table 5.28: Nominal update rate encoding

Encoded data				Nominal update rate (transmissions per minute)
nr_4	nr_3	nr_2	nr_1	nr
0	0	0	0	1
0	0	0	1	2
0	0	1	0	3
0	0	1	1	4
0	1	0	0	5
0	1	0	1	6
0	1	1	0	8
0	1	1	1	Invalid
1	0	0	0	10
1	0	0	1	12
1	0	1	0	15
1	0	1	1	20
1	1	0	0	30
1	1	0	1	60
1	1	1	0	0
1	1	1	1	Special

Requirement reference	
5.2.16.1.6	The 27-bit destination address (d) shall be the 27-bit address of the destination station for whom reservations are being created.
	Autotune reservation burst format
5.2.16.1.7	A directed request reservation burst with pr_flag = 0 shall indicate an autotune reservation.
5.2.16.1.8	Additional reservation data shall be set in accordance with table 5.29 with subfields defined in accordance with table 5.30.

Table 5.29: Encoding of additional data in autotune reservation burst

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
directed timeout (dt)	n - 11	dt_4	dt_3	dt_2	dt_1	f_{12}	f_{11}	f_{10}	f_9
frequency (f)	n - 10	f_8	f_7	F_6	f_5	f_4	f_3	f_2	f_1
length (lg)	n - 9	res	res	res	res	lg_4	lg_3	lg_2	lg_1
transmit control (trmt)	n - 8	res	res	trmt	res	do_{12}	do_{11}	do_{10}	do_9
directed offset (do)	n - 7	do_8	do_7	do_6	do_5	do_4	do_3	do_2	do_1
override flag (or); receiver control (rcvr); nominal update rate (nr); pr_flag = 0	n - 6	or	$rcvr_2$	$rcvr_1$	0	nr_4	nr_3	nr_2	nr_1
NOTE: "res" refers to bits available for the information field.									

Table 5.30: Directed request reservation field encoding

Subfield	Range	Encoding	Definitions
length (lg)	0 to 15	See table 5.24	lg is one less than the number of slots that are reserved.
directed timeout (dt)	0 to 15	A value of 15 cancels the reservation	dt = the number of planned future transmissions reserved in slots spaced M1 slots apart.
nominal rate (nr)	0 to 60	See table 5.28 When pr_flag = 0, nr = special is invalid	See table 5.28.

Subfield	Range	Encoding	Definitions
override flag (or)	0 to 1	See clause 5.2.16.3.1	or indicates whether the current directed request reservation burst overrides all previous directed request reservations issued by the station on the indicated frequency.
receiver control (rcvr)	0 to 3	00 = Station must continue to monitor the current frequency; 01 = Station must monitor the indicated frequency; 10 = Autonomous decision; 11 = Station must continue to monitor the current frequency and also the indicated frequency	Defines handling of receiver tuned to frequency used to receive this burst.
transmit control (trmt)	0 to 1	0 = cancel transmissions on the current frequency (see clauses 5.2.10.5.27 to 5.2.10.5.28) 1 = continue transmission on the current frequency	
directed offset (do)	0 or 2 to 2^{12} - 1	do = 1: invalid	do = 0 implies directed rate reservation. do > 1 implies directed slot reservation. For do > 1, do = the first slot in which to transmit.
offset to first reserved slot (off)	2 to 2^9-1	off = 0,1: invalid	off = the first slot in which to transmit (for plea response)
additional slots (a_j)	1 to 2^k - 1 (k = 6,12)	a_j = 20 hex and nr not equal to special: invalid NOTE: k is the number of bits in each a_j. k = 6 for nr not equal to "special", and k = 12 for nr = "special". j is the number of additional slots.	For nr not equal to "special", a_j is encoded as two's complement offset about a nominal slot defined by the offset to the first slot, and the nominal rate. For nr = special, a is encoded as a binary increment from the previously-reserved slot. a_j refers to the additional slot.
frequency (f)	See table 5.26	See table 5.26	Defines new frequency for transmissions of required data.
plea response flag (pr_flag)	See clause 5.2.16.1.2		

Requirement reference	
5.2.16.1.9	A reservation with do = 0, rcvr = 00 binary and f <> current frequency is invalid and shall be handled as per clause 5.2.5.
	Plea response burst format
5.2.16.1.10	A directed request reservation with pr_flag = 1 shall indicate a network entry plea response.
5.2.16.1.11	In this case, the reservation data not previously defined shall be encoded as indicated in tables 5.31 and 5.32 with subfields set in accordance with table 5.30, consisting of: a) the offset to a first reserved slot; and b) offsets to an additional n reserved slots as appropriate.
NOTE: This protocol is intended as a response for a VSS user which has no knowledge of the reservation table and must be given a large number of reservations in a single transmission. It is primarily intended as a response to a plea for help during rapid network entry (e.g. when a peer station asks for help in setting up a sequence of streams for synchronization bursts). However, it may be transmitted by the destination station to give itself a large number of reservations quickly.	

Table 5.31: Encoding of additional data with nr <> "special"

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
additional slots (a_j)	n - 15	$a_{11,6}$	$a_{11,5}$	$a_{8,6}$	$a_{8,5}$	$a_{8,4}$	$a_{8,3}$	$a_{8,2}$	$a_{8,1}$
	n - 14	$a_{11,4}$	$a_{11,3}$	$a_{7,6}$	$a_{7,5}$	$a_{7,4}$	$a_{7,3}$	$a_{7,2}$	$a_{7,1}$
	n - 13	$a_{11,2}$	$a_{11,1}$	$a_{6,6}$	$a_{6,5}$	$a_{6,4}$	$a_{6,3}$	$a_{6,2}$	$a_{6,1}$
	n - 12	$a_{10,6}$	$a_{10,5}$	$a_{5,6}$	$a_{5,5}$	$a_{5,4}$	$a_{5,3}$	$a_{5,2}$	$a_{5,1}$
	n - 11	$a_{10,4}$	$a_{10,3}$	$a_{4,6}$	$a_{4,5}$	$a_{4,4}$	$a_{4,3}$	$a_{4,2}$	$a_{4,1}$
	n - 10	$a_{10,2}$	$a_{10,1}$	$a_{3,6}$	$a_{3,5}$	$a_{3,4}$	$a_{3,3}$	$a_{3,2}$	$a_{3,1}$
	n - 9	$a_{9,6}$	$a_{9,5}$	$a_{2,6}$	$a_{2,5}$	$a_{2,4}$	$a_{2,3}$	$a_{2,2}$	$a_{2,1}$
	n - 8	$a_{9,4}$	$a_{9,3}$	$a_{1,6}$	$a_{1,5}$	$a_{1,4}$	$a_{1,3}$	$a_{1,2}$	$a_{1,1}$
offset to first reserved slot (off)	n - 7	$a_{9,2}$	$a_{9,1}$	off_9	off_8	off_7	off_6	off_5	off_4
nominal rate(nr); pr_flag = 1	n - 6	off_3	off_2	off_1	1	nr_4	nr_3	nr_2	nr_1

Table 5.32: Encoding of additional data for nr = "special"

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
additional slots (a_j)	n - 15	res	res	res	res	$a_{5,12}$	$a_{5,11}$	$a_{5,10}$	$a_{5,9}$
	n - 14	$a_{5,8}$	$a_{5,7}$	$a_{5,6}$	$a_{5,5}$	$a_{5,4}$	$a_{5,3}$	$a_{5,2}$	$a_{5,1}$
	n - 13	$a_{4,8}$	$a_{4,7}$	$a_{4,6}$	$a_{4,5}$	$a_{4,4}$	$a_{4,3}$	$a_{4,2}$	$a_{4,1}$
	n - 12	$a_{4,12}$	$a_{4,11}$	$a_{4,10}$	$a_{4,9}$	$a_{3,12}$	$a_{3,11}$	$a_{3,10}$	$a_{3,9}$
	n - 11	$a_{3,8}$	$a_{3,7}$	$a_{3,6}$	$a_{3,5}$	$a_{3,4}$	$a_{3,3}$	$a_{3,2}$	$a_{3,1}$
	n - 10	$a_{2,8}$	$a_{2,7}$	$a_{2,6}$	$a_{2,5}$	$a_{2,4}$	$a_{2,3}$	$a_{2,2}$	$a_{2,1}$
	n - 9	$a_{2,12}$	$a_{2,11}$	$a_{2,10}$	$a_{2,9}$	$a_{1,12}$	$a_{1,11}$	$a_{1,10}$	$a_{1,9}$
	n - 8	$a_{1,8}$	$a_{1,7}$	$a_{1,6}$	$a_{1,5}$	$a_{1,4}$	$a_{1,3}$	$a_{1,2}$	$a_{1,1}$
offset to first reserved slot (off)	n - 7	res	res	off_9	off_8	off_7	off_6	off_5	off_4
nominal rate (nr); pr_flag = 1	n - 6	off_3	off_2	off_1	1	1	1	1	1
NOTE: "res" refers to bits available for the information field.									

Requirement reference	
5.2.16.1.12	Additional reserved slots shall be encoded as follows:
5.2.16.1.13	Slots 1 to n shall be encoded in additional slots a_1 to a_n;
5.2.16.1.14	Additional slots a_{n+1} to a_N, where N is the maximum number of additional slots that can be accommodated in the formats defined by tables 5.31 and 5.32, shall be set to zero.
NOTE: In the format defined by table 5.31, up to 11 additional slots can be accommodated with offsets encoded using 6 bits. In the format defined by table 5.32, up to 5 additional slots can be accommodated with offsets encoded using 12 bits.	

5.2.16.2 Directed request parameters

Requirement reference	
5.2.16.2.1	The directed request protocol shall implement the system parameters defined in table 5.33.

Table 5.33: Directed request VSS system parameters

Symbol	Parameter Name	Minimum	Maximum	Recommended default	Increment
V52	Minimum response delay	1 slot	500 slots	20 slots	1 slot

Requirement reference	
5.2.16.2.2	The VSS user shall provide the destination address and any of the parameters V52 and Quality of Service parameters (Q2a to Q2d, Q4 and Q5) for which the default values are not desired.
	Parameter V52 (minimum response delay)
5.2.16.2.3	Parameter V52 shall be the minimum time that a station will provide to a responder in order to ensure timely delivery in case a retransmission is required.

5.2.16.3 Directed request reception procedures

Requirement reference	
	Autotune reception procedures
5.2.16.3.1	Upon receipt of a burst containing an autotune reservation (pr_flag = 0), the station shall update its reservation table and carry out the actions as specified in table 5.34.

Table 5.34: Action on receipt of an autotune reservation burst

Directed offset (do)	Directed timeout (dt)	Action
0	any	Operate autonomously.
1	any	Invalid
1 < do < M1	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 nr - 1, the slots equal to truncate (do + (k x M1/nr) + j x M1) through (lg + truncate (do + (k x M1/nr) + j x M1)) after the first slot of the received burst.
1 < do < M1	dt = 15	Reserve the following slots for the destination to broadcast: for k equal to 0 to nr - 1, the slots equal to truncate (do + (k x M1/nr)) through (lg + truncate (do + (k x M1/nr))) after the first slot of the received burst.
do > M1-1	any	Invalid.

Requirement reference	
5.2.16.3.2	If the override (or) flag is set to 1, the destination shall cancel all previously placed autotune reservations made by the source station on frequency f (see clause 5.2.10.5).
5.2.16.3.3	Otherwise, the station shall retain the previous reservations.
5.2.16.3.4	The burst is invalid, and shall be handled as per clause 5.2.5, if the frequency subfield fails to map to a known frequency, or indicates a frequency on which the transmitter cannot transmit.
	Plea response reception procedures
5.2.16.3.5	Upon receipt of a burst containing a plea response reservation (pr_flag = 1), a station shall reserve the slots equal to "off" after the first slot of the received burst and the series of slots r_j for the destination to broadcast.
5.2.16.3.6	If nr <> "special", then r_j shall be: r_j = (off + truncate(j × M1/nr) + a_j) for j = 1 to n, where n is the number of additional slots defined in the additional slots subfield (see clause 5.2.16.1.13).
5.2.16.3.7	If nr = "special", then r_j shall be defined as: r_j = (off + [sum from m = 1 to j] a_m) for j = 1 to n.
NOTE 1: In the case of directed time out (dt) = 15, slots are reserved in the current superframe for the destination to transmit periodic broadcast reservation bursts with periodic offset (po) = 0 and periodic time out (pt) = 0. These bursts have the effect of cancelling the directed reservations for each slot (see clause 5.2.16.4 and clause 5.2.16.5). NOTE 2: Only the destination station cancels previously placed reservations (e.g. due to a directed request with the override bit set). Other stations wait until the destination station announces its intent before updating their reservation tables. NOTE 3: Burst length is not included in the plea response, so these reservations are for single slots only.	

5.2.16.4 Directed request transmission procedures

Requirement reference	
	Recommendation
5.2.16.4.1	Void
5.2.16.4.2	Void
5.2.16.4.3	Void
	Autotune transmission procedures
5.2.16.4.4	Void
5.2.16.4.5	Void
	Retransmission after no response
5.2.16.4.6	There shall be no automatic retransmission of plea response bursts (pr_flag = 1).
5.2.16.4.7	Void
5.2.16.4.8	Void
	Cancellation of autotune reservation
5.2.16.4.9	Void
5.2.16.4.10	Void
	Plea response transmission procedures
5.2.16.4.11	A station transmitting a plea response (pr_flag = 1) shall set the destination (d) to the destination of the burst, the offset (off) subfield to the offset from the first slot of the reservation burst to the first slot in which to transmit, and the nominal rate (nr) subfield to the nominal number of times per M1 slots that a synchronization burst is to be sent on the frequency used for transmission.
5.2.16.4.12	The value of the offset (off) subfield shall be greater than V52.
5.2.16.4.13	A station shall ensure that the slots selected in the transmission satisfy the nominal update rate requirements and all of the requirements of clause 5.2.6.2.
5.2.16.4.14	A station shall check to determine if a previous plea response had been sent to the mobile making the plea (i.e. the destination ID for this plea response).
5.2.16.4.15	If a previous plea response had been sent to the mobile making the plea, the station shall begin the list of reserved slots with the remaining (future) reservations from the earlier plea response.
NOTE: When the plea response cannot encode sufficient reservations for a full 60 s, the destination station may issue a subsequent plea as the initial set of reservations is consumed.	

5.2.16.5 Directed request response procedures

Requirement reference	
	Response to an autotune reservation with directed offset (do) greater than one (do > 1)
5.2.16.5.1	On receiving an autotune reservation transmission (plea response flag pr_flag = 0) with the directed offset (do) subfield greater than one (do > 1), the responder station as indicated by the destination address shall set the TV11 (reservation hold timer) timer (see clause 5.2.10.2) equal to the value of the directed time out (dt) subfield for each of the slots indicated in the autotune reservation transmission.
5.2.16.5.2	The responder station shall transmit in each of the reserved slots.
5.2.16.5.3	Each response burst shall contain the periodic broadcast reservation field, with periodic offset (po) = 0 and the periodic time out (pt) = [min(3, TV11 − 1)], unless a general failure is being transmitted.
5.2.16.5.4	After transmission, the timer TV11 shall be decremented.
5.2.16.5.5	When TV11 reaches zero (TV11 = 0), the responder shall not transmit a response to the directed request.
5.2.16.5.6	Upon cessation of directed transmissions, the responder shall resume default autonomous behaviour on the GSCs, reserving new slots as required (see notes 1 and 2).
	Recommendation
5.2.16.5.7	To simplify and ease the reversion from a) directed slot operations on local channels, to b) directed rate or autonomous mode operations on another channel, ground stations should attempt to autotune mobile stations (using a directed slot reservation) to the new channel, for a period of at least 60 s, prior to release.
5.2.16.5.8	The mobile station should then use the BND reservation to continue the stream while it is building the reservation table and operating under ground control.
	Response to an autotune reservation with directed offset equal to zero
5.2.16.5.9	If the directed offset (do) subfield is equal to zero (do = 0), the responder station shall operate autonomously by using the periodic broadcast procedures with the nominal periodic rate (V11) set to the nominal update rate (nr) in the autotune reservation transmission for the next (dt × M1) slots, with the a/d bit set to zero.
5.2.16.5.10	Upon cessation of directed transmissions, the responder shall resume default autonomous behaviour on the GSCs, reserving new slots as required (see notes 3 and 4).
	Cancellation of autonomous periodic broadcasts
5.2.16.5.11	If transmit control (trmt) = 0 and if the responder was transmitting autonomously the VSS user data for which a directed request reservation has been received, then the responder shall cancel its existing reservations in accordance with 5.2.10.5.27 and 5.2.10.5.28, and operate in accordance with the parameters of the directed request.
5.2.16.5.12	If transmit control (trmt) = 1 and if the responder was transmitting autonomously on the channel on which a directed request reservation was received, then the responder shall maintain its autonomous transmissions on the channel.
	Cancellation of directed request broadcasts
5.2.16.5.13	If the responder receives a directed request reservation burst with the directed time out (dt) subfield set to 15 (dt = 15), then it shall cancel its existing reservations in accordance with 5.2.10.5.27 and 5.2.10.5.28.
	Response if unable to support directed request
5.2.16.5.14	If the responder is unable to support the directed request, it shall transmit a GENERAL FAILURE (see clause 5.2.20.1) with error type (err) = 05 hex.
5.2.16.5.15	If there are insufficient receiver resources to meet the request, it shall set bit 1 of the parameter subfield to one.
5.2.16.5.16	All other bits in the parameter subfield shall be set to zero.

	Recommendation
5.2.16.5.17	If possible, a responder sending a GENERAL FAILURE should use slots reserved by the ground station on the channel as indicated in the directed request.
5.2.16.5.18	Otherwise, the station should use the combined periodic/incremental reservation protocol to place the transmission on the channel on which the directed request was received.
NOTE 1: As an example, when a station completes its series of directed synchronization burst transmissions on a local channel, it will resume autonomous synchronization bursts on the GSCs. This behaviour allows a ground station to implicitly hand off a mobile as the mobile departs a defined airspace, and it also ensures appropriate behaviour if the mobile unexpectedly flies out-of-coverage or the ground station fails. NOTE 2: When a station is no longer required to transmit a response to the directed request, it will revert to the autonomous mode. NOTE 3: As an example, when a station completes its series of directed synchronization burst transmissions on a local channel, it will resume autonomous synchronization bursts on the GSCs. This behaviour allows a ground station to implicitly hand off a mobile as the mobile departs a defined airspace, and it also ensures appropriate behaviour if the mobile unexpectedly flies out-of-coverage or the ground station fails. NOTE 4: When a station is no longer required to transmit a response to the directed request, it will revert to the autonomous mode.	

5.2.17 Block reservation protocols specification

5.2.17.1 Superframe block reservation burst format

Requirement reference	
5.2.17.1.1	A reservation ID (rid) = 0, an extended reservation ID (erid) = 00010, and reservation fields set in accordance with table 5.35, with subfields defined in accordance with table 5.36, shall indicate a superframe block reservation.
NOTE: The superframe block reservation protocol establishes a series of blocks of slots in which no other station is allowed to place a reservation or to transmit. The second frame block reservation protocol establishes a block at the beginning of each UTC second. Network entry transmissions are also prohibited (see clause 5.2.6.4.3) in both types of blocks. The superframe block reservation protocol provides a facility for rebroadcasting of the block reservation by a mobile.	

Table 5.35: Superframe block reservation bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
destination address (d)	n - 10	d_{24}	d_{23}	d_{22}	d_{21}	d_{20}	d_{19}	d_{18}	d_{17}
	n - 9	d_{16}	d_{15}	d_{14}	d_{13}	d_{12}	d_{11}	d_{10}	d_9
	n - 8	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1
block length (blg)	n - 7	blg_5	blg_4	blg_3	blg_2	blg_1	d_{27}	d_{26}	d_{25}
re-broadcast offset (roff)	n - 6	$roff_8$	$roff_7$	$roff_6$	$roff_5$	$roff_4$	$roff_3$	$roff_2$	$roff_1$
block repeat rate (br)	n - 5	res	res	res	res	br_4	br_3	br_2	br_1
block start (bs)	n - 4	bs_8	bs_7	bs_6	bs_5	bs_4	bs_3	bs_2	bs_1
block offset (bo)	n - 3	bo_8	bo_7	bo_6	bo_5	bo_4	bo_3	bo_2	bo_1
extended reservation ID (erid), block timeout (bt)	n - 2	0	0	0	1	0	res	bt_2	bt_1
NOTE: "res" refers to bits available for the information field.									

Table 5.36: Superframe reservation field encoding

Subfield	Range	Encoding	Definitions
block timeout (bt)	0 to 3		bt x M1 = the number of slots for which the block reservation should be maintained.
block repeat rate (br)	1 to 60	See table 5.28. Codes 0111, 1110 and 1111 are invalid	Defines the number of blocks per minute.
re-broadcast offset (roff)	2 to 255	roff = 0,1 invalid	roff indicates the slot in which the re-broadcast transmission should be made.
block start (bs)	2 to 255	bs = 0,1 invalid	bs identifies a slot relative to the transmission slot which is the first slot of the first reserved block.
block offset (bo)	-127 to +127	Two's complement math	bo identifies an offset of each reserved block at a future time defined by bt x M1.
block length (blg)	0 to 31		blg is one less than the number of slots reserved for the block.
destination address (d)	See table 5.24	Ignored if ro = bs and octets (n - 10) through to (n - 8) are available for use within the information field.	d is the 27-bit address of the destination station which is required to re-broadcast the blocking message.

Requirement reference	
5.2.17.1.2	The information field shall extend up to the last nine octets prior to the CRC.

5.2.17.2 Second frame block reservation burst format

Requirement reference	
5.2.17.2.1	A reservation ID (rid) = 0, an extended reservation ID (erid) = 00011, and reservation fields set in accordance with table 5.37, with subfields defined in accordance with table 5.38, shall indicate a second frame block reservation.

Table 5.37: Second frame block reservation bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
timeout (vt)	n - 3	vt_6	vt_5	vt_4	vt_3	vt_2	vt_1	sz_5	sz_4
size (sz)	n - 2	0	0	0	1	1	sz_3	sz_2	sz_1

Table 5.38: Second frame block reservation field encoding

Subfield	Range	Definitions
size (sz)	0 to 31	Number of slots to block after the start of each UTC second.
timeout (vt)	1 to 60	Value of TV61.

Requirement reference	
5.2.17.2.2	The information field shall extend up to the last octet prior to the CRC.

5.2.17.3 Void

5.2.17.4 Superframe block reservation reception procedures

Requirement reference	
5.2.17.4.1	Void.
5.2.17.4.2	Upon receipt of a burst containing a superframe block reservation, the station shall update its reservation table and carry out the actions as specified in tables 5.39a and 5.39b.
NOTE: The actions in table 5.39b reserve the slot used by the station to provide a superframe block reservation in subsequent superframes.	

Table 5.39a: Actions on receipt of a superframe block reservation burst

Block start (bs)	Block offset (bo)	Block timeout (bt)	Action
bs < 2	Any	Any	Invalid
bs ≥ 2	Any except 0	0, 1, 2	Reserve the following slots for the source or a mobile directed by the source to broadcast: if bt = 1 or 2 then for j equal to 0 to bt and k equal to 0 to br - 1, the slots equal to truncate (bs + (k × M1 / br) + j × M1) through (blg + truncate (bs + (k × M1 / br) + j × M1)) after the first slot of the received burst AND for j equal to bt + 1 to 3 and k equal to 0 to br - 1, the slots equal to truncate (bs + bo + (k × M1 / br) + j × M1) through (blg + truncate (bs + bo + (k × M1 / br) + j × M1)) after the first slot of the received burst
bs ≥ 2	0	0, 1 or 2	Reserve the following slots for the source or a mobile directed by the source to broadcast: for j equal to 0 to bt and k equal to 0 to br - 1, the slots equal to truncate (bs + (k × M1 / br) + j × M1) through (blg + truncate (bs + (k × M1 / br) + j × M1)) after the first slot of the received burst Thereafter, terminate the reservations.
bs ≥ 2	Any	3	Reserve the following slots for the source or a mobile directed by the source to broadcast: for j equal to 0 to bt and k equal to 0 to br - 1, the slots equal to truncate (bs + (k × M1 / br) + j × M1) through (blg + truncate (bs + (k × M1 / br) + j × M1)) after the first slot of the received burst
NOTE: The actions defined in table 5.39a establish a series of reserved blocks of slots.			

Table 5.39b: Further actions on receipt of a superframe block reservation burst

Block offset (bo)	Block timeout (bt)	Action
Any except 0	0, 1, 2	Reserve the following slots for the source to broadcast: if bt = 1 or 2 then for j equal to 1 to bt, the slot equal to (j × M1) after the first slot of the received burst AND for j equal to bt + 1 to 3, the slot equal to (bo + j × M1) after the first slot of the received burst
0	0, 1 or 2	Reserve the following slots for the source to broadcast: for j equal to 1 to bt, the slot equal to (j × M1) after the first slot of the received burst Thereafter, terminate the reservations.
Any	3	Reserve the following slots for the destination to broadcast: for j equal to 1 to bt, the slot equal to (j × M1) after the first slot of the received burst.
NOTE: The actions defined in table 5.39b reserve the slot used by the station to provide a superframe block reservation in subsequent superframes.		

Requirement reference	
5.2.17.4.3	If the re-broadcast offset (roff) is not equal to the block start (bs), the station shall carry out the actions specified in table 5.39c.
NOTE: The actions in table 5.39c are carried out if a ground station wishes to re-broadcast the block message via a mobile.	

Table 5.39c: Action on receipt of a superframe block reservation burst if roff is not equal to bs

Re-broadcast offset (roff)	Block offset (bo)	Block timeout (bt)	Action
roff < 2	Any	Any	Invalid
roff ≥ 2	Any except 0	0, 1, 2	Reserve the following slots for the destination to broadcast: if bt = 1 or 2 then for j equal to 0 to bt, the slot equal to (roff + j × M1) after the first slot of the received burst AND for j equal to bt + 1 to 3, the slot equal to (roff + bo + j × M1) after the first slot of the received burst.
roff ≥ 2	0	0, 1 or 2	Reserve the following slots for the destination to broadcast: for j equal to 0 to bt, the slot equal to (roff + j × M1) after the first slot of the received burst. Thereafter, terminate the reservations.
roff ≥ 2	Any	3	Reserve the following slots for the destination to broadcast: for j equal to 0 to bt, the slot equal to (roff + j × M1) after the first slot of the received burst.
NOTE: The actions defined in table 5.39c are carried out if a ground station wishes to rebroadcast the block message via a mobile.			

5.2.17.5 Second frame block reservation parameters

Requirement reference	
5.2.17.5.1	The VSS user shall provide a value for the parameter TV61, defined in table 5.40, for which the default values are not desired.

Table 5.40: Second frame block reservation parameters

Symbol	Parameter Name	Minimum	Maximum	Default	Increment
TV61	Second frame block reservation timeout	1 superframe	60 superframes	4 superframes	1 superframe
V66	Second frame block size	0 slots	31 slots	8 slots	1 slot

Requirement reference	
5.2.17.5.2	Void
	Timer TV61 (second frame block reservation timeout)
5.2.17.5.3	The timer TV61 shall control the time which a second frame block reservation is valid.
5.2.17.5.3a	When timer TV61 times out, the mobile station shall return to the initial state defined in table 5.40.
	Parameter V66 (second frame block size)
5.2.17.5.4	Parameter V66 shall be the size of the second frame block.
5.2.17.5.5	Void.
NOTE: There is one V66 parameter for each channel.	

5.2.17.6 Second frame block reservation reception procedures

Requirement reference	
5.2.17.6.1	Upon receipt of a burst containing a second frame block reservation, the station shall update its reservation table by reserving the first sz slots of every UTC second on the channel on which the second block reservation was received, and set the timer TV61.

5.2.17.7 Void

5.2.17.7a Procedures for responding to a rebroadcast request for superframe block reservation

Requirement reference	
5.2.17.7a.1	On receipt of a superframe block reservation directed to itself, station B shall transmit a superframe block reservation in the directed slot.
5.2.17.7a.2	The station shall set the block repeat rate (br), block offset (bo) and block timeout (bt) subfields to the values contained in the received burst.
5.2.17.7a.3	The station shall set the block start (bs) so as to indicate the same first slot of the first block as was directed in the received superframe block reservation, calculated relative to the first slot of the transmission.
5.2.17.7a.4	The station shall set the re-broadcast offset (roff) subfield equal to the block start (bs) subfield and shall not include a destination subfield.

5.2.18 Response protocol specification

5.2.18.1 Response burst format

Requirement reference	
5.2.18.1.1	A reservation ID (rid) = 0 with extended reservation ID (erid) = 00000binary with reservation fields set in accordance with table 5.41 shall indicate a response burst.

Table 5.41: Response burst reservation bit encoding

Description	Octet	Bit Number							
		8	7	6	5	4	3	2	1
destination address (d)	n - 5	d_{24}	d_{23}	d_{22}	d_{21}	d_{20}	d_{19}	d_{18}	d_{17}
	n - 4	d_{16}	d_{15}	d_{14}	d_{13}	d_{12}	d_{11}	d_{10}	d_9
	n - 3	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1
reservation ID	n - 2	0	0	0	0	0	d_{27}	d_{26}	d_{25}

Requirement reference	
5.2.18.1.2	In the case that the address type field (see clause 5.3.1.2) is equal to 7, bits 1 through 24 of the destination subfield (d) shall be absent.
5.2.18.1.3	In this case the information field shall extend up to the last one octet prior to the CRC.
5.2.18.1.4	Otherwise, the destination subfield (d) shall be the 27-bit address of the destination station (for which the response is addressed).
5.2.18.1.5	In this case the information field shall extend up to the last four octets prior to the CRC.
5.2.18.1.6	No reservation shall be made as a result of receiving a response burst.
5.2.18.1.7	The VSS user shall provide the destination address.
NOTE 1: In the typical application, the response will be broadcast; however, some applications might require a unicast response. NOTE 2: The response burst with the address type field set equal to seven is intended as an alternative to the null reservation type defined in clause 5.2.9.	

5.2.19 General request protocol specification

5.2.19.1 General request burst format

Requirement reference	
5.2.19.1.1	To request a peer station to transmit a particular burst, a station shall send the burst described in table 5.42 to the desired destination station.

Table 5.42: General request bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
	5	r-mi$_5$	. . . . . .			r-mi$_1$	0	0	1
requested message ID (r-mi)	6	x	r-mi$_n$	. . . . . .					r-mi$_6$
VSS user specific parameter (prm)	7 to n - 3	. . . . . .							prm$_1$
	n - 2	prm$_n$	. . . . . .						

	. . . .		Denotes variable length field

Requirement reference	
5.2.19.1.2	VSS user-specific parameters shall be encoded starting in the octet following the most significant (high order) bit of the r-mi field.
5.2.19.1.3	Unused bits (x) shall be filled with 0 on transmit and ignored on receive.
5.2.19.1.4	The values of the subfields shall be computed as defined in table 5.43.

Table 5.43: General request field encoding

Subfield	Range	Encoding	Notes
requested message ID (r-mi)	See clause 5.2.2.5		
VSS user specific parameter (prm)			This is an optional field defined by the VSS user.

Requirement reference	
5.2.19.1.5	The requested message ID (r-mi) shall define the VSS user, in accordance with table 5.4, which is responsible for handling the request.

5.2.19.2 General request procedures

Requirement reference	
	Requester action
5.2.19.2.1	For a VSS user to request that a peer VSS user transmit certain information, the VSS user shall transmit a general request burst with the requested ID (r-mi) field set to the desired response.
5.2.19.2.2	The unicast request reservation field shall be used if a single response is required from a single station.
5.2.19.2.3	Void.
	Responder action
5.2.19.2.4	The addressed responder(s) shall respond in the reserved slots with the requested bursts except for the conditions specified in clause 5.2.6.4.
	Exceptional cases
5.2.19.2.5	If the requested function is not supported, the responder shall send a general failure (see clause 5.2.20) with the ok bit set to zero, the requested ID (r-mi) field set to the requested VSS user, the backoff delay (bd) set to FF hex, and the error type (err) set to either 00 hex or 80 hex.
5.2.19.2.6	If the requested function is supported, but there has been no response in time for transmission in the reserved slot, then the responder shall transmit a general failure (see clause 5.2.20) with the ok bit set to zero, the requested ID and extended ID fields set to the requested VSS user, the backoff delay set to an estimate of when the data will be available, and the error type set to either 7E hex or FE hex.
5.2.19.2.7	Void.
5.2.19.2.8	Void.

5.2.20 General response protocol specification

5.2.20.1 General response burst format

Requirement reference	
5.2.20.1.1	A station shall transmit a general response burst (either a General Failure or General Confirm) as defined in table 5.44 with the parameters defined in table 5.45 in response to certain requests from another station as described below.

Table 5.44: General response bit encoding

Description	Octet	Bit number							
		8	**7**	**6**	**5**	**4**	**3**	**2**	**1**
confirm/failure flag (ok)	5	ok	1	1	1	0	1	0	1
requested message ID (r-mi)	6	$r\text{-}mi_k$	....						$r\text{-}mi_1$
backoff delay (bd)	7	bd_8	bd_7	bd_6	bd_5	bd_4	bd_3	bd_2	bd_1
error type (err)	8	err_8	err_7	err_6	err_5	err_4	err_3	err_2	err_1
VSS user specific parameter (prm)	9 to n - 3	....							prm_1
	n - 2	prm_n	....						

	. . . .		Denotes variable length field

Table 5.45: General response field encoding

Subfield	Range	Encoding	Notes
confirm/failure flag (ok)		1 = General confirm 0 = General failure	
requested message ID (r-mi)	See clause 5.2.2.4		r-mi can extend into bits 1 to 4 of octet 7 for extended message IDs. In this case, bd starts in bit 5 of octet 7 and other fields are shifted later in the message by 4 bits.
backoff delay (bd)	0 to 255	integer seconds, FF hex = forever	In seconds, ignore on confirm.
error type (err)	See table 5.46		
VSS user specific parameter (prm)	Defined by the VSS user		

Requirement reference	
5.2.20.1.2	The requested message ID (r-mi) shall indicate the identity of the peer VSS user to which a response is being generated.
5.2.20.1.3	The general response burst shall include one of the following reservation fields: unicast request reservation or response reservation.
5.2.20.1.4	The destination subfield contained in the reservation field shall indicate which VSS user is being responded to.
5.2.20.1.5	If the ok bit is set to 1 (i.e. a General Confirm), and the response does not utilize the parameter field, the information field shall contain the requested message ID (r-mi) subfield only with the remaining parameters omitted.
5.2.20.1.6	If the ok bit is set to 1 and the parameter field is used, then the bd and err fields shall be included and set to 00 hex.
5.2.20.1.7	If the ok bit is set to zero (i.e. a General Failure), then the remaining parameters shall define the reason why the request failed.
5.2.20.1.8	Error type (err) shall be encoded in accordance with table 5.46.
5.2.20.1.9	Error types 00 hex to 7F hex shall apply to the responding station.
5.2.20.1.10	Error types 80 hex to FF hex shall apply to the responding system.
NOTE: A mobile VSS user receiving a ground station-based error type (err) from one ground station may immediately transmit the same request to another ground station of the same ground system.	

Table 5.46: Error type definition

Cause	Function	Parameter Encoding (prm bits 1 to 8)							
		8	7	6	5	4	3	2	1
00 hex	Unsupported local function. The parameters (defining the protocol options supported) will be filled in when defined.	0	0	0	0	0	0	0	0
01 hex	Out of local resources.	Reserved.							
02 hex	VSS user-specific local error.	Defined by the VSS user.							
03 hex	Terrestrial network not available.	Reserved. Set to zero on transmit, ignore on receipt.							
04 hex	Terrestrial network congestion.								
05 to 7D hex	Reserved.								
7E hex	No response from VSS user.								
7F hex	Other unspecified local reason.								
80 hex	Unsupported global function. The parameters (defining the protocol options supported) will be filled in when defined.	0	0	0	0	0	0	0	0
81 hex	Out of global resources.	Reserved.							
82 hex	VSS user-specific global error.	Defined by the VSS user.							
83 to FC hex	Reserved.	Reserved. Set to zero on transmit, ignore on receipt.							
FD hex	Rejected for internal policy reasons								
FE hex	No response from VSS user.								
FF hex	Other unspecified system reason.								

5.2.20.2 General response procedures

Requirement reference	
5.2.20.2.1	If a reservation has been placed for a response or acknowledgement but the VSS sublayer has not received the response or acknowledgement from the VSS user in time for the scheduled reservation, the station shall send a General Failure (see clause 5.2.20.1.7) with cause code 7E hex or FE hex.
5.2.20.2.2	If a response is received, the VSS shall inform the VSS user.

5.2.21 Retransmission procedures

Requirement reference	
5.2.21.1	After transmitting a burst containing a reservation for a peer station (i.e. unicast request reservation, directed request reservation, information transfer request reservation) and not receiving a response by the expected slot, a station shall either retransmit the request or inform the VSS user and LME if $Q5_{num}$ attempts have already been made or if more than $Q5_{wait}$ seconds have elapsed since the VSS user initiated the request.
5.2.21.2	The re-transmitting station shall wait for $Q5_{min} + min(U(x), Q5_{max})$ seconds before attempting to retransmit the burst, where: **U(x)** is a uniform random number generated between 0 and x; **x** is defined by $Q5_{mult} \times (Q5_{exp}^{retrans}) \times M1 /(M1+ 1\text{-}u)$; **u** is the number of occupied slots within the past minute on the channel concerned; **retrans** is the number of times that a burst has been retransmitted (see note).
NOTE: If $Q5_{num} = 1$, no re-transmission is attempted and hence parameters $Q5_{max}$, $Q5_{min}$, $Q5_{mult}$, $Q5_{exp}$ are not used.	

5.3 DLS sublayer

5.3.1 Services

5.3.1.1 General

Requirement reference	
5.3.1.1.1	The DLS shall support broadcast and multicast connectionless communications.
NOTE 1: It is intended that NSCOP be used for air/ground (A/G) communications and ZOCOP for air/air (A/A) communications. NOTE 2: Apart from the procedures for link set-up and tear down, the NSCOP and ZOCOP protocols operate identically. NOTE 3: Any two stations have one DLE pair per frequency.	

5.3.1.2 Data transfer

Requirement reference	
5.3.1.2.1	LME data shall be transferred in the information fields of UDATA data link protocol data units (DLPDUs).
5.3.1.2.2	LME data shall be contained in UCTRL DLPDUs only.
NOTE 1: The Frame Mode Subnetwork Dependent Convergence Function (SNDCF) may concatenate multiple packets, but this is presented as a single user data packet to the DLS. NOTE 2: UDATA DLPDUs consist of UINFO DLPDUs for broadcast of user data packets, and UCTRL DLPDUs for broadcast of LME data. UDATA is the broadcast equivalent of DATA and embraces all broadcast-type DLPDUs.	

5.3.1.3 Station address encoding

Requirement reference	
	Address type
5.3.1.3.1	The address type field shall be encoded as defined in table 5.47.

Table 5.47: Address type field encoding

Bit encoding			Description type	Bits 1 to 24
27	**26**	**25**		
0	0	0	Mobile	Non-unique identity
0	0	1	Aircraft	24-bit ICAO address
0	1	0	Ground vehicles	Nationally administered address space
0	1	1	Reserved	Future use
1	0	0	Ground station	ICAO-administered address space
1	0	1	Ground station	ICAO-delegated address space
1	1	0	Reserved	Future use
1	1	1	All stations broadcast	All stations

Requirement reference	
	Non-unique identity address
5.3.1.3.2	Void.
5.3.1.3.2a	A mobile station using the non-unique identity address shall randomly choose a 24-bit address.
5.3.1.3.2b	The non-unique identity address of all zeros shall not be used.
5.3.1.3.2c	The non-unique identity address of all ones shall be used as a destination address for broadcast applications only.
5.3.1.3.2d	All radio units located at a station shall use the same non-unique identity address.
5.3.1.3.2e	If the station detects that another station is using the same random address, it shall stop transmitting on the current address;
5.3.1.3.2f	it shall then randomly select a new address that is not already present in its PECT.
5.3.1.3.2g	It shall used this new address in subsequent transmissions.
5.3.1.3.3	When using VDL Mode 4 for ATS applications, aircraft shall use the unique 24-bit ICAO address (see note 2).
	Aircraft specific address
5.3.1.3.4	The aircraft specific address field shall be the 24-bit ICAO aircraft address.
	ICAO-administered ground station specific addresses
5.3.1.3.5	Void.
5.3.1.3.6	Void.
	ICAO-delegated ground station specific addresses
5.3.1.3.7	Void.
	Broadcast and multicast addresses
5.3.1.3.8	The broadcast and multicast addresses shall be used only as a destination address for UDATA DLPDUs.
	Broadcast and multicast address encoding
5.3.1.3.9	The broadcast and multicast addresses shall be encoded as in table 5.48.
NOTE 1: The processing of ambiguous data resulting from use of the non-unique address is an end system issue. NOTE 2: Bits 25, 26, and 27 of the full 27-bit address will be 1, 0, and 0 respectively, as is defined in table 5.47. NOTE 3: Mobiles using non-unique addresses cannot be unambiguously identified and this mode of use is itself inherently insecure. Higher level functions are required when there is a need to provide data origin authentication in case non-unique addresses are used. NOTE 4: Non-unique addressing violates the subnetwork requirements of the ICAO ATN, and mobiles using this addressing mode cannot implement ATN applications.	

Table 5.48: Broadcast and multicast address encoding

Broadcast destination	Type field	Specific address field
All mobiles that use non-unique addresses	000	All ones
All mobiles	001	All ones
All ground stations of a particular provider	100 or 101, as necessary	Most significant bits: Variable length Provider code Remaining bits: All ones
All ground stations with ICAO-administered addresses	100	All ones
All ground stations	101	All ones
All stations	111	All ones

5.3.1.4 DLS burst formats

NOTE: The functionality defined here only supports reception of UDATA DLPDUs by a mobile.

Requirement reference	
	UDATA DLPDU definition
5.3.1.4.1	A UDATA shall consist of one of the UDATA bursts defined in tables 5.49 to 5.51 with the VSS user supplied QoS and reservation parameters.
5.3.1.4.2	The burst format shall be one of tables 5.49, 5.50, or 5.51 based on the UDATA ID (udid) of the message as defined by table 5.52.
5.3.1.4.3	A UCTRL shall have ucd set to 0 and have the appropriate ud field set to the value of ucid per table 5.52.
5.3.1.4.4	A UINFO shall have ucd set to 1 and have the appropriate ud field set to the value of uinf per table 5.53.

Table 5.49: One-byte UDATA burst format

Description	Octet	Bit Number							
		8	**7**	**6**	**5**	**4**	**3**	**2**	**1**
Message ID, UDATA ID (ud1)	5	$ud1_5$	$ud1_4$	$ud1_3$	$ud1_2$	$ud1_1$	ucd	1	1
UDATA DLPDU	6	Information field							
	7								
	8								
	9								
	10								

Table 5.50: Two byte UDATA burst format

Description	Octet	Bit Number							
		8	**7**	**6**	**5**	**4**	**3**	**2**	**1**
Message ID	5	1	1	1	1	0	ucd	1	1
UDATA ID (ud2)	6	$ud2_8$	$ud2_7$	$ud2_6$	$ud2_5$	$ud2_4$	$ud2_3$	$ud2_2$	$ud2_1$
UDATA DLPDU	7	Information field							
	8								
	9								
	10								
	11								

Table 5.51: Three-byte UDATA burst format

Description	Octet	Bit Number							
		8	**7**	**6**	**5**	**4**	**3**	**2**	**1**
Message ID	5	1	1	1	1	1	ucd	1	1
UDATA ID (ud3)	6	$ud3_{16}$	$ud3_{15}$	$ud3_{14}$	$ud3_{13}$	$ud3_{12}$	$ud3_{11}$	$ud3_{10}$	$ud3_9$
	7	$ud3_8$	$ud3_7$	$ud3_6$	$ud3_5$	$ud3_4$	$ud3_3$	$ud3_2$	$ud3_1$
UDATA DLPDU	8	Information field							
	9								
	10								
	11								
	12								
NOTE: The UDATA DLPDU field may be up to ND4 octets long.									

Table 5.52: Encoding of the UDATA ID (udid) value

UDATA ID (udid)	Encoded by
0 to 29	table 5.49, ud1 = udid
30 to 285	table 5.50, ud2 = udid - 30
286 to 65 821	table 5.51, ud3 = udid - 286
NOTE: The UCTRL ID (ucid) subfield is defined in clause 5.4.2.6.	

Table 5.53: UINFO ID (uinf) assignments

UINFO ID (uinf)	Assignment
0 to 60 000	Reserved for future use
60 001 to 65 821	Messages reserved for transmission by ground station only and defined by ground station operator

5.3.2 DLS system parameters

Requirement reference	
5.3.2.1	The parameters needed by the DLS sublayer shall be as listed in table 5.54.

Table 5.54: Data link service system parameters

Symbol	Parameter name	Minimum	Maximum	Default	Increment
ND4	Maximum length of UDATA burst	23 octets	496 octets	271 octets	1 octet

5.3.2.1 Parameter ND4 (maximum length of a UDATA burst)

Requirement reference	
5.3.2.1.1	The parameter ND4 shall define the maximum size in octets of a UDATA burst.

5.3.3 DLS procedures

5.3.3.1 Broadcast

Requirement reference	
5.3.3.1.1	Only UDATA DLPDUs shall be broadcast (as opposed to other types of DLPDUs).
	Action on receipt of UDATA DLPDU
5.3.3.1.2	A station receiving a UDATA DLPDU shall forward the contents of the information field to the DLS user and take no further action.
NOTE:	Either the mobile or ground station may send user data packets at any time and is considered peers with respect to management of the link.

5.3.3.2 DLS not supported

Requirement reference	
5.3.3.2.1	If the responder to a DLS DLPDU other than a UDATA DLPDU does not support the DLS, then it shall transmit a general failure (see clause 5.2.20) with an error type of 80 hex in the slot reserved by the unicast request reservation field contained in the data DLPDU transmission.
NOTE:	The response upon receipt of a general failure, described in clause 5.2.20.1, is to not transmit another DLS burst to the ending station for the duration of the backoff timer.

5.3.3.3 User data packet reception

Requirement reference	
5.3.3.3.1	When a UDATA DLPDU is received without errors from another station, it shall be passed to the service user as a single incoming user data packet.
5.3.3.3.2	Otherwise it shall be discarded.
	Unacknowledged DLPDUs
5.3.3.3.3	UDATA DLPDUs shall be unacknowledged.

5.3.3.4 CTRL DLPDU

Requirement reference	
5.3.3.4.1	The CTRL DLPDU shall be used for the LME to establish and maintain links as defined in clause 5.4.

5.4 Link Management Entity (LME) sublayer

5.4.1 Services

Requirement reference	
5.4.1.1	The LME shall support the provision of broadcast services.

5.4.2 Synchronization burst format

5.4.2.1 General

Requirement reference	
5.4.2.1.1	All VDL Mode 4 stations shall transmit synchronization bursts to support link management.

5.4.2.2 Fixed and variable data fields

Requirement reference	
5.4.2.2.1	The synchronization burst shall consist of two portions: a fixed data field containing information that is sent with each synchronization burst and a variable data field containing additional system management information that does not need to be included in each synchronization burst.
NOTE 1: The variable data field can also include VSS user-specific information. NOTE 2: The fixed data field contains 55 bits of data consisting of bits 2 through 8 of octet 5 and all of octets 6 through 11 inclusive (the fixed data field begins after the source address (s) and message identification (mi) fields which consist of the first 4 octets and bit 1 of octet 5). The variable portion contains 54 bits of data consisting of octets 12 through 17 and bits 3 through 8 of octet 18. NOTE 3: Certain variable information fields have been assigned and are described in EN 302 842-3 [9].	

5.4.2.3 Fixed data field format

Requirement reference	
5.4.2.3.1	Stations shall have fixed data fields as defined in table 5.55.

Table 5.55: Synchronization burst format

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
TCP/SVQ change flag (tqc) baro/geo altitude (b/g) CPR Format even/odd (cprf) navigation integrity category (nic)	5	nic_4	nic_3	nic_2	nic_1	cprf	b/g	tqc	0
latitude (lat)	6	lat_8	lat_7	lat_6	lat_5	lat_4	lat_3	lat_2	lat_1
base altitude (balt)	7	$balt_{12}$	$balt_{11}$	$balt_{10}$	$balt_9$	lat_{12}	lat_{11}	lat_{10}	lat_9
	8	$balt_8$	$balt_7$	$balt_6$	$balt_5$	$balt_4$	$balt_3$	$balt_2$	$balt_1$
longitude (lon)	9	lon_8	lon_7	lon_6	lon_5	lon_4	lon_3	lon_2	lon_1
time figure of merit (tfom)	10	$tfom_2$	$tfom_1$	lon_{14}	lon_{13}	lon_{12}	lon_{11}	lon_{10}	lon_9
data age (da) information field ID (id)	11	da_4	da_3	da_2	da_1	id_4	id_3	id_2	id_1
ID extension 1 (id1) ID extension 2 (id2)	12	$id1_4$	$id1_3$	$id1_2$	$id1_1$	$id2_4$	$id2_3$	$id2_2$	$id2_1$
ID extension 3 (id3)	13	$id3_4$	$id3_3$	$id3_2$	$id3_1$	in_k	....		
information field (in)	14	....							
	15	....							
	16	....							
	17	in_{14}	in_{13}	in_{12}	in_{11}	in_{10}	in_9	in_8	in_7
	18	in_6	in_5	in_4	in_3	in_2	in_1		
NOTE: Mobile and ground stations are identified by the address type field in the 27-bit code (see clause 5.3.1.2).									

. . . . Denotes variable length field

Requirement reference	
5.4.2.3.2	The subfields shall be computed as defined in table 5.56.

Table 5.56: Synchronization burst field encoding (fixed data field)

Subfield	Range	Encoding	Notes
TCP/SVQ change flag (tqc)	Boolean	Encoded as described in clause 3.6 of ICAO VDL4 Technical Manual [1] if the message ID (see table 5.3) indicates that the burst is a directed synchronization burst. Otherwise reserved for future definition and set equal to 1.	
time figure of merit (tfom)	0 to 3	0 = primary certified 1 = primary/non-certified 2 = secondary 3 = none of the above	See clause 5.1.4.
navigation integrity category (nic)	0 to 15	See table 5.57 Values 12 to 15 are reserved for future definition.	
latitude (lat)	-90 to + 90°	12-bit low-resolution encoding according to the CPR encoding algorithm adapted for VDL Mode 4, as described in clause 5.6.	The 12-bit CPR encoding provides position to a resolution of approximately ±140 m, within a segment (patch) of approximately 600 NM.

Subfield	Range	Encoding	Notes
longitude (lon)	-180°to + 180°	14-bit low-resolution encoding according to the CPR encoding algorithm adapted for VDL Mode 4, as described in clause 5.6.	The 14-bit CPR encoding provides position to a resolution of approximately ±120 m, within a segment (patch) of approximately 600 NM.
CPR format even/odd	0 to 1	0 = even 1 = odd The CPR flag shall apply to all CPR encoded sub-fields included in the synchronization burst.	
base altitude (balt)	0 to 4 095	Base altitude is reported as specified in table 5.58.	
baro/geo altitude (b/g)	0 to 1	0 = barometric 1 = geometric	Indicates whether barometric or geometric base altitude is reported.
data age (da)	0 to 15	See table 5.59.	
information field ID (id)	0 to 15	As defined by application standards. Some values for the information field ID are pre-reserved and defined in clause 5.4.5.	Provides the information field identity contained in the variable data field (see clause 5.4.2.4).
ID extension	0 to 15	See clauses 5.4.2.3.3 to 5.4.2.3.9 below.	Provides a means of increasing the number of variable fields that can be accommodated.
information field (in)	-	As defined by application standards.	The information field contained in the variable data field (see clause 5.4.2.4).

Requirement reference	
5.4.2.3.3	The information field ID (id) and ID extension (idn) subfields shall provide addresses for information fields (in) as follows:
5.4.2.3.4	1) An information field ID (id) equal to F hex shall indicate that no information field is present.
5.4.2.3.5	2) An information field ID (id) subfield equal to 0 hex to 9 hex or B hex to E hex shall indicate one of 14 information fields of length 54 bits.
5.4.2.3.6	3) ID extension 1 (id1) subfield shall only be present if the information field ID (id) is equal to A hex.
5.4.2.3.7	4) An ID extension 1 (id1) subfield equal to 0 hex to 9 hex or B hex to F hex shall indicate one of 15 information fields of length 50 bits.
5.4.2.3.8	5) An ID extension n (idn) subfield shall only be present if the ID extension n - 1 (idn - 1) subfield is equal to A hex.
5.4.2.3.9	6) An ID extension n (idn) subfield equal to 0 hex to 9 hex or B hex to F hex shall indicate one of 15 information fields of length 54 - 4n bits.
5.4.2.3.10	The station shall encode its navigation integrity (nic) in accordance with table 5.57.

Table 5.57: Encoding of position Navigation Integrity Category (NIC)

NIC	Required Navigation Performance (RNP) class	Horizontal and Vertical containment radius (R_c)
0	Unknown integrity	$R_c \geq$ 20 NM
1	RNP-10	$R_c <$ 20 NM
2	RNP-4	$R_c <$ 8 NM
3	RNP-2	$R_c <$ 4 NM
4	RNP-1	$R_c <$ 2 NM
5	RNP-0,5	$R_c <$ 1 NM
6	RNP-0,3	$R_c <$ 0,6 NM
7	RNP-0,1	$R_c <$ 0,2 NM
8	RNP-0,05	$R_c <$ 0,1 NM
9	Undefined	$R_c <$ 75 m
10	Undefined	$R_c <$ 25 m
11	Undefined	$R_c <$ 7,5 m
12	Reserved for future definition	
13	Reserved for future definition	
14	Reserved for future definition	
15	Reserved for future definition	

Requirement reference	
5.4.2.3.11	The station shall encode base altitude in accordance with table 5.58.

Table 5.58: Base altitude encoding

Actual base altitude of transmitting station (feet)	Transmitted value of altitude	Decoded base altitude (feet) (geo = WGS84 height except as noted)
Unknown	0	altitude unknown
altitude < -1 305	1	less than -1 300
-1 305 ≤ altitude < -1 295	2	-1 300
-1 295 ≤ altitude < -1 285	3	-1 290
↓	↓	↓
-15 ≤ altitude < -5	131	-10
-5 ≤ altitude < 5	132	0
5 ≤ altitude < 15	133	10
↓	↓	↓
7 995 ≤ altitude < 8 005	932	8 000
8 005 ≤ altitude < 8 015	933	8 010
8 015 ≤ altitude < 8 037,5	934	8 025
8 037,5 ≤ altitude < 8 062,5	935	8 050
8 062,5 ≤ altitude < 8 087,5	936	8 075
↓	↓	↓
71 912,5 ≤ altitude < 71 950	3 490	71 925
71 950 ≤ altitude < 72 050	3 491	72 000
72 050 ≤ altitude < 72 150	3 492	72 100
72 150 ≤ altitude < 72 250	3 493	72 200
72 250 ≤ altitude < 72 350	3 494	72 300
72 350 ≤ altitude < 72 450	3 495	72 400
↓	↓	↓
129 950 ≤ altitude < 130 050	4 071	130 000
130 050 ≤ altitude	4 072	more than or equal to 130 100
	4 073 to 4 094	reserved
station on ground	4 095	station at 0 AGL

Requirement reference	
5.4.2.3.12	The data age (da) subfield shall be encoded based on the report latency which shall be the difference between the time of validity of the horizontal position data (latitude and longitude) and the time of transmission, in accordance with table 5.59.

Table 5.59: Report latency encoding and decoding

Report latency (ms)	Transmitted value of data age (da)	Decoded latency (ms)
difference < 100	0	50
100 ≤ difference < 200	1	150
200 ≤ difference < 300	2	250
↓	↓	↓
900 ≤ difference < 1 000	9	950
1 000 ≤ difference < 1 200	10	1 100
1 200 ≤ difference < 1 500	11	1 350
1 500 ≤ difference < 2 000	12	1 750
2 000 ≤ difference < 3 000	13	2 500
3 000 ≤ difference < 4 000	14	3 500
4 000 ≤ difference or unknown	15	unknown

Requirement reference	
5.4.2.3.13	If the report latency is greater than 4 s, then nic shall be set to 0.

5.4.2.4 Variable data field format

Requirement reference	
5.4.2.4.1	The variable data field shall be available to carry additional information as may be required by another VSS user or application.
5.4.2.4.2	The content and format of the variable data field shall be identified by the information field ID (id).
5.4.2.4.3	The format of the variable data field corresponding to a given id shall be as specified in the appropriate application standard.
NOTE: Up to 15 different information(in) fields can be addressed. Information fields 1 to 9 are given priority over other information fields. Further short information fields may be addressed with the use of the extension field ID of 10 (see clause 5.4.2.2 and table 5.56).	

5.4.2.5 Synchronization burst request

Requirement reference	
5.4.2.5.1	To request that a station transmit a synchronization burst with a specific information field, a station shall transmit a general request burst to the appropriate application process (see clause 5.4.5.2).

5.4.2.6 Broadcast link management burst

Requirement reference	
5.4.2.6.1	Broadcast link management data shall be contained within the information field of a UCTRL DLPDU as defined in table 5.60.

Table 5.60: UCTRL DLPDU burst format

Description	Octet	Bit Number							
		8	7	6	5	4	3	2	1
UCTRL DLPDU header	n	$ucid_5$	$ucid_4$	$ucid_3$	$ucid_2$	$ucid_1$	1	0	0
CTRL parameter 1: Parameter ID	n+1	id_8	id_7	id_6	id_5	id_4	id_3	id_2	id_1
Parameter length	n+2	lg_8	lg_7	lg_6	lg_5	lg_4	lg_3	lg_2	lg_1
Parameter value	n+3	$q1_8$	$q1_7$	$q1_6$	$q1_5$	$q1_4$	$q1_3$	$q1_2$	$q1_1$
	to	to							
	n+2 + lg	qlg_8	qlg_7	qlg_6	qlg_5	qlg_4	qlg_3	qlg_2	qlg_1
		more CTRL parameters							

Requirement reference	
5.4.2.6.2	ucid = 0 shall indicate that the UCTRL DLPDU is a GSIF containing any of the CTRL parameters defined in clause 5.4.3.
5.4.2.6.3	ucid = 1 shall indicate that the UCTRL DLPDU is a GSIF containing a CTRL DOS parameter only as defined in clause 5.4.3.
5.4.2.6.4	ucid = 5 to 31 are currently unassigned and available for future definition.

5.4.3 Control (CTRL) parameter formats

5.4.3.1 Encoding

Requirement reference	
5.4.3.1.1	The CTRL parameters described in this clause shall be included in the user data field of UCTRL DLPDUs.
5.4.3.1.2	Except for parameter ID00 (which must appear last), the parameters in a CTRL DLPDU shall be listed in non-decreasing numeric order.
NOTE 1: The tables in this clause are divided into three major columns that define the field name, the bit encoding and brief explanatory notes. NOTE 2: The CTRL parameters are extracted from the DLS sub-layer CTRL DLPDUs and sent to the LME.	

5.4.3.2 VDL Mode 4 parameter identification

Requirement reference	
5.4.3.2.1	Bits 7 and 8 of the parameter ID field shall allow simple identification of the purpose of the parameter as defined in table 5.61.

Table 5.61: VDL Mode 4 parameter identifier purpose

Bit 8	Bit 7	Purpose
0	0	General purpose information parameter
0	1	Ground-initiated modification parameter
1	0	Mobile-initiated information parameter
1	1	Ground-initiated information parameter

5.4.3.3 Ground-initiated modification parameters

Requirement reference	
5.4.3.3.1	A ground LME shall use the ground-initiated modification parameters to change the value of various parameters, for the frequency on which they are received, in one or more mobiles.
5.4.3.3.1a	Reception of these parameters shall replace any default settings for the same parameters, any settings resulting from a previous ground-initiated modification command, and any settings resulting from a previous channel management command (see EN 302 842-3 [9]), for the frequency on which they are received.
5.4.3.3.1b	Mobile LMEs shall not send a CTRL containing these parameters.
	VSS sublayer parameter
5.4.3.3.2	This parameter defines the value of VS1, VS2, VS4 and VS5 that a mobile shall use, encoded as per table 5.62.

Table 5.62: VSS sublayer parameter encoding

Field	Octet	Bit position								Notes
Parameter ID	n+1	0	1	0	0	0	0	0	1	VSS sublayer parameter
Parameter length	n+2	0	0	0	0	0	0	1	1	
Parameter value	n+3	$VS5_4$	$VS5_3$	$VS5_2$	$VS5_1$	$VS1_4$	$VS1_3$	$VS1_2$	$VS1_1$	VS1, VS5 slots
	n+4	0	0	$VS2_6$	$VS2_5$	$VS2_4$	$VS2_3$	$VS2_2$	$VS2_1$	VS2 (dB)
	n+5	0	$VS4_7$	$VS4_6$	$VS4_5$	$VS4_4$	$VS4_3$	$VS4_2$	$VS4_1$	VS4 (NM)

Table 5.62a: VSS sublayer parameter field encoding

Subfield	Range	Encoding	Notes
VS1	0 to 15 slots	0 to 15	
VS2	6 to 60 dB	0 to 54	
VS4	0 to 1000 NM	0 to 100	
VS5	1 to 16 slots	0 to 15	

Requirement reference	
	Quality of service parameter
5.4.3.3.3	This parameter defines the quality of service parameters that a mobile shall use for priority levels Q1min to Q1max, encoded as per table 5.63.

Table 5.63: Quality of service parameter encoding

Field	Octet	Bit position								Notes
Parameter ID	n+1	0	1	0	0	0	0	1	0	Quality of Service parameter
Parameter length	n+2	lg_8	lg_7	lg_6	lg_5	lg_4	lg_3	lg_2	lg_1	
Parameter value	n+3	$Q1max_4$	$Q1max_3$	$Q1max_2$	$Q1max_1$	$Q1min_4$	$Q1min_3$	$Q1min_2$	$Q1min_1$	
	n+4	$Q2a_8$	$Q2a_7$	$Q2a_6$	$Q2a_5$	$Q2a_4$	$Q2a_3$	$Q2a_2$	$Q2a_1$	Q2a (NM)
	n+5	$Q2b_8$	$Q2b_7$	$Q2b_6$	$Q2b_5$	$Q2b_4$	$Q2b_3$	$Q2b_2$	$Q2b_1$	Q2b (NM)
	n+6	$Q2c_8$	$Q2c_7$	$Q2c_6$	$Q2c_5$	$Q2c_4$	$Q2c_3$	$Q2c_2$	$Q2c_1$	Q2c (NM)
	n+7	$Q2d_8$	$Q2d_7$	$Q2d_6$	$Q2d_5$	$Q2d_4$	$Q2d_3$	$Q2d_2$	$Q2d_1$	Q2d (NM)
	n+8	$Q2d_{10}$	$Q2d_9$	$Q2c_{10}$	$Q2c_9$	$Q2b_{10}$	$Q2b_9$	$Q2a_{10}$	$Q2a_9$	Q2a, Q2b, Q2c, Q2d
	n+9	0	0	0	$Q4_5$	$Q4_4$	$Q4_3$	$Q4_2$	$Q4_1$	Q4

Table 5.63a: Quality of service parameter field encoding

Subfield	Range	Encoding	Notes
Q1min	0 to 14	0 to 14 (15 invalid)	
Q1max	0 to 14	0 to 14 (15 invalid)	
Q2a	0 to 1 000 NM	0 to 1 000 (1 001 to 1023 invalid)	
Q2b	0 to 1 000 NM	0 to 1 000 (1 001 to 1023 invalid)	
Q2c	0 to 1 000 NM	0 to 1 000 (1 001 to 1 023 invalid)	
Q2d	0 to 1000 NM	0 to 1 000 (1 001 to 1 023 invalid)	
Q4	1 to 20	0 to 19 (20 to 31 invalid)	

Requirement reference	
	m2 filter parameters
5.4.3.3.4	Table 5.64 shall define the encoding of parameters for the m2 filter (see ICAO VDL4 Technical Manual [1]) to be used by a mobile.

Table 5.64: m2 filter parameter encoding

Field	Octet	Bit position								Notes
Parameter ID	n+1	0	1	0	0	0	0	1	1	m2 filter parameters
Parameter length	n+2	0	0	0	0	0	0	1	1	
M2inc parameter value	n+3	i_8	i_7	i_6	i_5	i_4	i_3	i_2	i_1	M2inc
M2limit parameter value	n+4	l_{16}	l_{15}	l_{14}	l_{13}	l_{12}	l_{11}	l_{10}	l_9	M2limit
	n+5	l_8	l_7	l_6	l_5	l_4	l_3	l_2	l_1	

Table 5.64a: m2 filter parameter field encoding

Subfield	Range	Encoding	Notes
M2inc	1 to 256	0 to 255	
M2limit	1 to 65 536	0 to 65 535	

Requirement reference	
5.4.3.3.5	M2inc shall be encoded as an 8-bit unsigned integer.
5.4.3.3.6	M2limit shall be encoded as a 16-bit unsigned integer.
	CG1 filter parameters
5.4.3.3.7	The values of parameters used by mobiles for the CG1 filter (see ICAO VDL4 Technical Manual [1]) shall be as defined in table 5.65.
5.4.3.3.8	CG1_plea shall be encoded as an 8-bit unsigned integer.
5.4.3.3.9	CG1_range shall be encoded as an 8-bit unsigned integer.
5.4.3.3.10	TL5 shall be encoded as an 8-bit unsigned integer.
5.4.3.3.11	CG1_inc shall be encoded as an 8-bit unsigned integer.
5.4.3.3.12	1/CG1_decay shall be encoded as an 8-bit unsigned integer.
5.4.3.3.13	CG1_limit shall be encoded as a 16-bit unsigned integer.

Table 5.65: CG1 filter parameter encoding

Field	Octet	Bit position								Notes
Parameter ID	n+1	0	1	0	0	0	1	0	0	CG1 filter parameters
Parameter length	n+2	0	0	0	0	0	1	1	1	
CG1_plea parameter value	n+3	p_8	p_7	p_6	p_5	p_4	p_3	p_2	p_1	CG1_plea
CG1_range parameter value	n+4	r_8	r_7	r_6	r_5	r_4	r_3	r_2	r_1	CG1_range
TL5 parameter value	n+5	t_8	t_7	t_6	t_5	t_4	t_3	t_2	t_1	TL5
CG1_limit parameter value	n+6	l_{16}	l_{15}	l_{14}	l_{13}	l_{12}	l_{11}	l_{10}	l_9	CG1_limit

Field	Octet	Bit position								Notes
	n+7	l_8	l_7	l_6	l_5	l_4	l_3	l_2	l_1	
CG1_inc parameter value	n+8	i_8	i_7	i_6	i_5	i_4	i_3	i_2	i_1	CG1_inc
1/CG1_decay parameter value	n+9	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1	1/CG1_decay

Table 5.65a: CG1 filter parameter field encoding

Subfield	Range	Encoding	Notes
CG1_plea	1 to 256	0 to 255	
CG1_range	0 to 255	0 to 255	
TL5	0,1 to 16	0 to 159 (160 to 255 invalid)	
CG1_limit	1 to 65 536	0 to 65 535	
CG1_inc	1 to 256	0 to 255	
1/CG1_decay	1/256 to 1	0 to 255	

Requirement reference	
	Maximum number of missed reservations parameter
5.4.3.3.13a	This parameter shall define the number of consecutive missed reservations that will be used to determine a station is unreachable.
5.4.3.3.13b	This parameter shall be sent by a ground LME, as required, to adjust the timeliness of the LEAVE event.
5.4.3.3.13c	The parameter shall be encoded as per table 5.65b.
5.4.3.3.13d	The parameters (L1 and TL3) shall be defined as per table 5.70a.
NOTE: The range of the L1 counter is 1 to 255. The value of 0 (L1=0) is invalid.	

Table 5.65b: L1 parameter encoding

Field	Octet	Bit number								Notes
		8	**7**	**6**	**5**	**4**	**3**	**2**	**1**	
Parameter ID	**n + 1**	0	1	0	0	0	1	1	0	
Parameter length	**n + 2**	0	0	0	0	0	0	0	1	
Parameter value	**n + 3**	$L1_8$	$L1_7$	$L1_6$	$L1_5$	$L1_4$	$L1_3$	$L1_2$	$L1_1$	L1 counter
	n + 4	0	0	0	$TL3_5$	$TL3_4$	$TL3_3$	$TL3_2$	$TL3_1$	TL3 timer
	n + 5	$TL4_8$	$TL4_7$	$TL4_6$	$TL4_5$	$TL4_4$	$TL4_3$	$TL4_2$	$TL4_1$	TL4 timer
NOTE: See EN 302 842-4 [10] clause 5.2.3.4 for the definition of parameter TL4.										

Requirement reference	
	Random access parameter
5.4.3.3.14	The random access parameter shall define p, VS3 and TM2 used within the random access protocol, encoded as per table 5.66.
NOTE: p is encoded as hexadecimal 00 (= decimal 1 / 256) to hexadecimal FF (= decimal 256 / 256).	

Table 5.66: Random access parameter encoding

Field	Octet	Bit position								Notes
Parameter ID	n+1	0	1	0	0	1	0	0	0	Random access parameter encoding
Parameter length	n+2	0	0	0	0	0	1	0	0	
Parameter value	n+3	p_8	p_7	p_6	p_5	p_4	p_3	p_2	p_1	p
	n+4	$VS3_8$	$VS3_7$	$VS3_6$	$VS3_5$	$VS3_4$	$VS3_3$	$VS3_2$	$VS3_1$	
	n+5	$VS3_{16}$	$VS3_{15}$	$VS3_{14}$	$VS3_{13}$	$VS3_{12}$	$VS3_{11}$	$VS3_{10}$	$VS3_9$	
	n+6	$TM2_8$	$TM2_7$	$TM2_6$	$TM2_5$	$TM2_4$	$TM2_3$	$TM2_2$	$TM2_1$	

Table 5.66a: Random access parameter field encoding

Subfield	Range	Encoding	Notes
p	1/256 to 1	0 to 255	
VS3	1 to 65 536	0 to 65 535	
TM2	20 to 4 500	0 to 224 (225 to 255 invalid)	

5.4.3.4 Ground-initiated information parameters

Requirement reference	
5.4.3.4.1	A ground LME shall use ground initiated information parameters to inform one or more mobile LMEs about that ground-based system's capabilities.
5.4.3.4.1a	Mobile LMEs shall not send these parameters.
	Directory of Service (DOS) parameter
5.4.3.4.2	The Directory of Service parameter shall be encoded as defined in table 5.67.

Table 5.67: Directory of service (DoS) message encoding

Field	Octet	Bit position								Notes
		8	7	6	5	4	3	2	1	
Parameter ID	n+1	1	1	0	0	0	1	0	1	DoS parameter
Parameter length	n+2	lg_8	lg_7	lg_6	lg_5	lg_4	Lg_3	lg_2	lg_1	length
parameter value	n+3	gsc	ai_3	ai_2	ai_1	ent_4	ent_3	ent_2	ent_1	entry number (ent), **current channel subfield** additional service info (ai); GSC flag (gsc)
	n+4	si_8	si_7	si_6	si_5	si_4	si_3	si_2	si_1	service information (si)
	n+5	res	res	res	res	$anum_4$	$anum_3$	$anum_2$	$anum_1$	application number (anum) res field absent if anum field is absent.
	n+6	a_{18}	a_{17}	a_{16}	a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	application 1 (a_1)
	n+7	to								
	n+8	a_{k8}	a_{k7}	a_{k6}	a_{k5}	a_{k4}	a_{k3}	a_{k2}	a_{k1}	application k (a_k)
	n+9	gsc	ai_3	ai_2	ai_1	f_{12}	f_{11}	f_{10}	f_9	**channel subfield:** additional service info (ai); GSC flag (gsc)
	n+10	f_8	f_7	f_6	f_5	f_4	f_3	f_2	f_1	frequency (f)
	n+11	si_8	si_7	si_6	si_5	si_4	si_3	si_2	si_1	service information (si)
	n+12	res	res	res	res	$anum_4$	$anum_3$	$anum_2$	$anum_1$	application number (anum)
	n+13	a_{18}	a_{17}	a_{16}	a_{15}	a_{14}	a_{13}	a_{12}	a_{11}	application 1 (a_1)
	n+14					to				
	n+15	a_{k8}	a_{k7}	a_{k6}	a_{k5}	a_{k4}	a_{k3}	a_{k2}	a_{k1}	application k (a_k)
	n+16	sit_6	sit_5	sit_4	sit_3	sit_2	sit_1	x	x	service information type (sit)
NOTE: Bits denoted "x" may be used within the reservation field. "res" refers to bits available for the information field.										

Requirement reference	
5.4.3.4.3	If the DOS parameter is included within a UCTRL DLPDU with the ucid subfield set to 1, then the DOS parameter ID and parameter length shall be omitted and no other parameter included in the UCTRL.
NOTE: This makes it possible to provide a single slot transmission of basic DoS information.	

Requirement reference	
5.4.3.4.4	The current channel subfield shall always be present.
5.4.3.4.5	One, two, or more other channel subfields (channel 1, channel 2 etc) shall be added as required in a continuous bit sequence.
5.4.3.4.6	The contents of the channel subfields shall be determined by the ai subfield as defined in table 5.68.

Table 5.68: Directory of service message subfield encoding

Subfield	Range	Encoding	Notes
entry number (ent)	0 to 15	ent = entry number of Directory of Services message.	up to 16 different DOS messages can be accommodated associated with each ground station transmitting DOS messages.
frequency (f)		See table 5.26 Absent in current channel subfield.	indicates the channel on which the DOS service is provided.
GSC flag (gsc)	0 to 1	set to 1 if channel is a GSC	
additional service information (ai)	0 to 7	bit 1: set to 1 if si field included bit 2: set to 1 if anum field present. bit 3: set to 1 if application (a) subfield(s) present.	
service information type (sit)	0 to 63	Defines services indicated by each bit in the service information field. As defined by application standards.	
service information (si)	Contains 8 single bit flags	Bits indicate the services provided on the indicated channel. bit set if service is available. Meaning of bits defined by application standards. field absent if ai bit 1 = 0.	
application number (anum)	0 to 15	Indicates the number of application fields present. field absent if ai bit 2 = 0.	
application (a)	0 to 255	Identifies a single service defined by application standards field absent if ai bit 3 = 0 if ai bit 1 = 0 and ai bit 2 $\neq$ 0, only one application subfield shall be present.	

Requirement reference	
5.4.3.4.7	The service information type (sit) subfield shall follow the last channel subfield.
5.4.3.4.8	The subfields within each channel subfield shall be computed as defined in table 5.68.
5.4.3.4.9	Each DOS parameter shall override any previous DOS parameter from the same ground station with the same entry number (ent).
5.4.3.4.10	The upper bit of the application field shall be used as an extension field, so that a 0 indicates a one octet field and a 1 indicates that the ID continues in the next octet.
5.4.3.4.11	Application fields shall be allocated as defined in table 5.69.

Table 5.69: Allocation of application fields

Encoding (decimal equivalent)	Allocation
0 to 3	defined for broadcast services (see EN 302 842-3 [9])
4 to 31	reserved for future allocation by ICAO
32 to 63	reserved for private allocation by service provider
64 to 127	reserved for future allocation by ICAO delegated authority
128 to 255	reserved for future use (extension of application field)

Requirement reference	
5.4.3.4.12	Service information type (sit) fields shall be allocated as defined in table 5.70.

Table 5.70: Allocation of service information type fields

Encoding (decimal equivalent)	Allocation
0 to 3	defined for broadcast services (see EN 302 842-3 [9])
4 to 31	reserved for future allocation by ICAO
32 to 47	reserved for private allocation by service provider
48 to 63	reserved for future allocation by ICAO delegated authority

5.4.3a LME timers and parameters

5.4.3a.1 General

Requirement reference	
5.4.3a.1.1	The LME service shall implement the system parameters defined in table 5.70a.

Table 5.70a: Management entity system parameters

Symbol	Parameter name	Minimum	Maximum	Default	Increment
L1	Maximum number of missed reservations	1	255	3	1
TL3	Inter-miss timer	0 s	31 s	5 s	1 s

5.4.3a.2 Counter L1 (maximum number of missed reservations) and Timer TL3 (inter-miss timer)

Requirement reference	
5.4.3a.2.1	Parameter L1 shall be the maximum number of missed reservations before a station assumes that a peer station is unreachable.
5.4.3a.2.2	There shall be one counter L1 per peer station as well as one per frequency (per peer station).
5.4.3a.2.3	Counter L1 (both the all-frequencies and the appropriate frequency elements of the counter) shall be set to zero when a transmission is received from a peer station.
5.4.3a.2.4	Counter L1 shall be incremented when no transmission is received from a peer station for which there was a prior reservation made by the peer station for itself, that is: (a) a periodic broadcast reservation; (b) an incremental broadcast reservation; (c) a combined periodic and incremental reservation; (d) a BND reservation; (e) a unicast request reservation with sdf = 1; (f) the reservation for the acknowledgement within an information transfer request reservation (see clause 5.3.1); (g) a plea response addressed to itself (see clause 5.4.4.3.6); if the particular L1 has not been incremented in the prior TL3 seconds.
5.4.3a.2.5	When the all-frequencies counter L1 exceeds the maximum number of missed reservations (the value of parameter L1), the peer station shall be marked as unreachable in the peer entity contact table (PECT) (see clause 5.4.4.2) and it shall attempt to handoff to another ground station following the procedures of EN 302 842-3 [9].
5.4.3a.2.6	Stations marked as unreachable shall be retained in the PECT for a period of time not less than 60 min.
5.4.3a.2.7	When the single-frequency counter L1 exceeds the maximum number of missed reservations, the peer station shall be marked unreachable on that frequency.
NOTE 1: The reason that L1 is not incremented when no response is received in a slot that another station had reserved for its peer is the possibility that the reservation itself was lost and thus no reliable inference can be made on the reachability of the peer station. Lost or missed reservation in this context means no decoded transmission, i.e. the receiving station has not decoded a transmission it was expecting in a slot reserved for the sending station. NOTE 2: PECT entries are retained for a period of time in order to stabilize acquisition and track re-initialization performance for stations at the limits of coverage, and also to support potential search and rescue applications that may be developed in the future. Reservations for unreachable stations are retained in the reservation table and allowed to expire normally. These reservations represent real transmissions that may be expected to occur at the indicated times.	

5.4.4 LME procedures

5.4.4.1 Synchronization burst procedures

Requirement reference	
5.4.4.1.1	All stations shall transmit the appropriate synchronization burst defined in clause 5.4.2 depending on whether it is a mobile station or a ground station.
5.4.4.1.2	If the synchronization burst is transmitted with a periodic broadcast protocol, it shall use default QoS parameters except as defined in table 5.71.
5.4.4.1.3	If the synchronization burst is not transmitted with a periodic broadcast protocol, slot selection shall use the default QoS parameters defined for the selected reservation protocol or user supplied QoS parameters.

Table 5.71: Synchronization burst parameters

Symbol	Parameter name	Default
V11	Nominal periodic rate	6
Q1	Priority	14
Q2a	Slot selection range constraint for level 1	380 NM
Q2b	Slot selection range constraint for level 2	380 NM
Q2c	Slot selection range constraint for level 3	0 NM
Q2d	Slot selection range constraint for level 4	380 NM
Q3	Replace queued data	TRUE

Requirement reference	
5.4.4.1.4	The values of the subfields shall be the latest available data that can be obtained by the station at the start of the slot that is a maximum of eight slots before the first slot of the intended transmission (see note).
5.4.4.1.5	Where time is used to calculate fields in the transmission, it shall be the time associated with the latitude and longitude data contained in the transmission.
	Transmission of synchronization bursts supporting applications
5.4.4.1.6	A station shall transmit synchronization bursts in accordance with a request from a peer station as described in clause 5.4.2.5, at the specified rate, and containing the information field corresponding to the requested information field ID (r-id).
5.4.4.1.6a	The station shall transmit additional synchronization bursts required to meet the demands of any application.
5.4.4.1.6b	In the event that an application request requiring the transmission of synchronization bursts is delivered by means of a directed request, the required bursts shall be transmitted in the slots reserved by the directed request protocol.
NOTE: The interaction between the LME and the application of specific requirements for transmission of synchronization bursts is a local issue.	

Requirement reference	
	Mobile stations
5.4.4.1.6c	Whenever a mobile station is not directed to transmit synchronization bursts on any frequency, it shall transmit mobile synchronization bursts (see clause 5.1.1.14 in EN 302 842-3 [9]) at least once per M1 slots on all GSCs which it can receive.
5.4.4.1.6d	When transmitting autonomously on the GSCs, a mobile station shall use the standard parameters defined in table 5.71.
	Ground stations
	Recommendation
5.4.4.1.7	Void
	Procedures for conflict resolution
5.4.4.1.8	For the purposes of assessing whether another reservation conflicts with a reservation for a synchronization burst, the station shall apply the procedures defined in clause 5.2.6.4.
5.4.4.1.9	In this case, the quality of service parameters defined in table 5.72 or user supplied parameters shall be applied to the synchronization burst reservation (see note 2).
NOTE 1: Upon termination of all directed synchronization burst transmissions on a channel(s) other than the GSCs, mobile stations may employ the network entry procedures, as described in clause 5.4.4.3, to quickly enter the GSC network(s). NOTE 2: These QoS parameters place a tighter constraint than the defaults for original slot selection, which would always result in a slot being selected. The tighter constraint forces the stream to dither to find slots that might be available at higher levels and hence reduces the probability of slot conflict. NOTE 3: This requirement is less stringent than that contained in the ICAO VDL4 Technical Manual [1], in order not to place undue restrictions on the processing speed required to construct the burst prior to transmission.	

Table 5.72: Synchronization burst parameters for conflict resolution

Symbol	Parameter name	Value
Q1	Priority	14 (unless otherwise specified in the information field)
Q2a	Slot selection range constraint for level 1	360 NM
Q2b	Slot selection range constraint for level 2	360 NM
Q2c	Slot selection range constraint for level 3	360 NM
Q2d	Slot selection range constraint for level 4	360 NM

5.4.4.2 Peer Entity Contact Table (PECT)

Requirement reference	
5.4.4.2.1	Every station shall maintain a Peer Entity Contact Table (PECT) of all known stations.
5.4.4.2.2	For each station for which a report has been received in the previous 15 minutes, the PECT table shall include as a minimum the station type (e.g. ground or mobile), the station ID, the station's last known position and altitude, the station's reservation information, the time of the last transmission, and an L1 counter.
5.4.4.2.2a	For each station for which a report has been received in the previous 60 minutes, the PECT table shall include as a minimum the station type (e.g. ground or mobile), the station ID, the station's last known position and altitude, and the time of the last transmission.
5.4.4.2.3	The ability to reach a peer station shall be assumed lost after L1 missed reservations.
	Handling transmissions received from other stations
5.4.4.2.4	A station shall be capable of handling in the PECT received transmissions from 2 047 stations.
	Recommendation
5.4.4.2.5	In order to be able to accommodate predictions on long term air traffic growth, a station should be capable of handling in the PECT received transmissions from 4 095 stations.
5.4.4.2.6	When there are too many stations to keep them all in the PECT, a station shall: a) delete stations from the PECT that are marked as unreachable prior to deleting reachable stations; b) maintain closer aircraft in the PECT in preference to more distant aircraft, while also weighting approaching aircraft over receding aircraft.
	Reports from targets with the same non-unique address
5.4.4.2.7	When performing a global CPR decode on a received report from a target with a non-unique address (which requires use of a previously stored PECT entry), a receiving station shall not use stored PECT data on receipt of reports that appear to be from multiple targets with the same non-unique address, which cannot be distinguished by any other means (see note 1).
	Recommendation
5.4.4.2.8	A station should record and track the movements of aircraft with non-unique addresses for a time that is sufficient to be able to distinguish them (see note 2).
5.4.4.2.9	A receiving station shall use local decode for 5 minutes after it is able to distinguish multiple targets with the same non-unique address which it had previously been unable to distinguish (see note 3).
	Reception processing capability
5.4.4.2.10	A station shall be capable of processing new transmissions received from previously unknown stations at a rate of at least (number of receivers installed) × 75 new transmissions per second.
	Slot map management
5.4.4.2.11	A station shall maintain a table of all reservations in the next 4 × M1 + 128 slots for all channels on which the station is transmitting and/or receiving.
5.4.4.2.12	The VDL Mode 4 transceiver shall be designed so as to ensure that under all expected operating conditions, it is capable of storing all reservation data required by clause 1.3.6.1 of the ICAO VDL4 Technical Manual [1].
5.4.4.2.13	In exceptional circumstances, if it is necessary to discard a reservation due to lack of available memory, the affected slot shall be flagged as not available, and shall not be selected by the station for a transmission.

NOTE 1: This is to ensure that a global CPR decode is not accidentally performed using the reports from two different targets with the same non-unique address, and means that only local CPR decode will be used in processing the received report.

NOTE 2: If the tracks of such stations can be followed with certainty, then stored PECT data may be used on receipt of the reports.

NOTE 3: A period of 5 minutes should allow sufficient time for the tracks of two or more aircraft with the same non-unique address to be distinguished.

5.4.4.3 Network entry protocol specifications

Requirement reference	
	Network entry parameters
5.4.4.3.a	The network entry protocol shall implement the system parameters defined in table 5.7.2a.
NOTE: There is a separate CG1 filter for each channel which implements the exposure filter.	

Table 5.72a: Plea parameters

Symbol	Parameter name	Minimum	Maximum	Recommended default	Increment
CG1_plea	Minimum pool size of peer stations for plea	1	256	2	1
CG1_range	Range limit used in digital filter for "exposure" determination	0 NM	255 NM	100 NM	1 NM
TL5	Maximum delay for plea response	0,1 s	16 s	2 s	0,1 sec
CG1_limit	Digital filter threshold which triggers network entry due to station being "exposed"	1	65 536	2 000	1
CG1_inc	Digital filter increment when a synchronization burst from a previously unknown station is detected	1	256	max{min[5 × (V11−1), 256], 1}	1
CG1_decay	Decay rate for CG1 filter	1 / 256	1	247 / 256	0
CG1_reach	Maximum unreachability time	1 min	15 min	3 min	1 min
TL6	Minimum time delay for CG1 filter	1 slot	255 slots	75 slots	1 slot

Requirement reference	
	Parameter CG1_plea (minimum pool size of peer stations for plea)
5.4.4.3.b	CG1_plea shall define the minimum number of viable peer stations which must be identified before a plea can be issued.
5.4.4.3.c	The plea target shall be selected uniformly from a set of CG1_plea viable peer stations.
	Parameter CG1_range (maximum range for exposure filter)
5.4.4.3.d	CG1_range shall define the threshold beyond which newly-identified peer stations are disregarded for purposes of determining "exposure".
NOTE 1: This digital filter is intended to trigger network re-entry in cases where a station identifies a large number of new peer stations in a short period of time (e.g. during climb-out from an airport or when crossing a mountain range). Conversely, since it is normal for new stations to be identified at great range, stations at great range should not be included in the determination of when a station has become "exposed". NOTE 2: If the CG1_range cannot be determined because a station is not aware of its own location, it is set to a default value of zero (CG1 = 0). Therefore, unless a service provider has specifically reset CG1_range = 0 by a previous command detected by the station, all newly identified peer stations will increment the value of the filter.	

Requirement reference	
	Parameter TL5 (maximum delay for plea response)
5.4.4.3.1	TL5 shall specify the maximum allowed time interval between receiving a plea and transmitting a plea response.
5.4.4.3.2	A station receiving a plea shall aim to generate a response at the earliest opportunity and before TL5 s have expired.
5.4.4.3.3	If a response cannot be generated in TL5 seconds, the station shall purge the plea and not respond (see note).
	Parameter CG1_limit (exposure filter threshold)
5.4.4.3.3a	When CG1 is greater than or equal to CG1_limit, the station shall consider itself "exposed" (i.e. it has recently detected a large number of users which were not previously in its PECT, indicating that its reservation table may be incomplete).
5.4.4.3.3b	Consistent with the requirements of clause 1.5.6.3 of ICAO VDL4 Technical Manual [1], the station shall then reset CG1 = 0, and perform network entry, on the affected channel.
	Parameter CG1_inc (exposure filter unknown station increment)
5.4.4.3.3c	CG1_inc shall be the value by which CG1 is incremented each time a station which has been unreachable for at least CG1_reach minutes or a previously unknown peer station is detected with a station-to-station range less than or equal to CG1_range, or with station-to-station range which is indeterminate.
	Parameter CG1_decay (decay rate for CG1 filter)
5.4.4.3.3d	CG1_decay shall be the per second decay rate for CG1.
	Recommendation
5.4.4.3.3e	CG1_decay should be selected such that CG1 represents an approximate metric for the uncertainty in reservation information associated with newly detected stations.
	Parameter CG1_reach
5.4.4.3.3f	CG1_reach shall be the maximum allowed time since a station became unreachable after expiration of L1 without increasing the CG1 filter in accordance with the procedures defined in table 5.72b.
NOTE: This is intended to avoid creating reservations that will not be used.	

Table 5.72b: Asynchronous events affecting value of management filter CG1

Events	Values of CG1
Station detects a synchronization burst from a station which has been unreachable on the channel for at least CG1_reach minutes or from a previously unknown station at a station-to-station range which can either be calculated as less than CG1_range or is indeterminate.	$CG1_{new} = CG1_{old} + CG1\_inc$
Station detects a synchronization burst from a station which has been unreachable on the channel for less than CG1_reach minutes or from a previously unknown station at a station-to-station range which can be calculated as greater or equal to CG1_range.	$CG1_{new} = CG1_{old}$
Once per second.	$CG1_{new} = truncate(CG1_{old} \times CG1\_decay)$

Requirement reference	
	Parameter TL6 (minimum time delay for plea)
5.4.4.3.3g	TL6 shall be the minimum time delay, measured in slot intervals, between the initiation of network entry procedures and the issuance of the first plea on the channel, and also the retry interval between successive examinations of the pool of viable peer stations for plea.
5.4.4.3.3h	TL6 shall be set when a station tunes to a new frequency on which it has less than nr reservations for synchronization bursts over the next M1 slots; or, upon expiration, if there are no viable peer stations for the plea.
5.4.4.3.3i	TL6 shall be cleared when the number of reservations for synchronization bursts, over the next M1 slots, equals or exceeds nr; or, when the station has monitored the frequency for M1 slots since the initiation of network entry procedures.
5.4.4.3.3j	Upon expiration of TL6, the station shall transmit a plea if a viable peer station exists.
	Conditions for application of network entry procedures
5.4.4.3.4	Void.
5.4.4.3.4a	When entering the network, a VSS user shall apply at least one of the network entry procedures defined in clauses 5.4.4.3.4h to 5.4.4.3.4r, 5.4.4.3.12b to 5.4.4.3.12i, or 5.4.4.3.13, under any of the conditions identified in table 5.72c, on the indicated channels (see note).
NOTE: When required to perform network entry, a station will apply at least one of the following network entry procedures: a) Network entry by listening to the channel for M1 + 126 slots followed by either: - full-slot random transmission with periodic, incremental, or combined periodic/incremental reservation types; - plea-to-self (i.e. reserving slots for itself to transmit); b) Network entry using plea/response procedures; c) Network entry via BND. The network entry period required by different applications may be determined in the future, when the applications are more clearly defined.	

Table 5.72c: Conditions for network entry

Condition	Channel(s)
Power ON	Global Signalling Channels (GSCs).
m2 ≥ M2_limit	Any channel on which the station intends to transmit synchronization bursts in autonomous mode.
CG1 ≥ CG1_limit	Any channel on which the station intends to transmit synchronization bursts in autonomous mode.
NOTE: The m2 test addresses retuning as well as short periods of receiver non-performance for reasons such as receiver deactivation or desensitization in a single-antenna configuration during transmission on another channel.	

Requirement reference	
5.4.4.3.4b	The events affecting the value of the CG1 filter shall be as defined in table 5.72b.
5.4.4.3.4c	If CG1 ≥ CG1_limit, CG1 shall be reset to 0.
5.4.4.3.4d	A VDL4 station shall re-enter the network as soon as possible after any of the conditions requiring network entry to be performed have been met (see note 1).
	Recommendation
5.4.4.3.4e	The station should aim to re-enter the network within x seconds after any of the conditions requiring network entry to be performed have been met.
5.4.4.3.4f	The default value of x shall be 10, but different values may be used depending on the applications used with VDL Mode 4 (see note 2).
5.4.4.3.4g	Following successful application of the network entry procedures of this clause on a given channel, yielding (60/nr) reservations, nr = nominal rate, a station shall not re-apply the procedures of this clause, on that channel, for a period of M1 slots.
NOTE 1:	Network entry will be performed if: (a) The m2 filter is exceeded (see clause 5.1.3.3), due either to: - Return to operation following power failure; - Return to operation following hardware or software failure; - Return to operation following receive function that is blocked by own station transmission on another channel (e.g. through a common antenna); - Return to operation following any other inability to receive. (b) The value of CG1 is greater than or equal to CG1_limit, which may be due to: - Return to operation following complete or partial RF shadowing for a period of time resulting in an inability to receive from other stations.
NOTE 2:	The value of 10 seconds has been chosen as the default for the following reasons (a) for the reporting rates typical of many applications involving regular position reporting by all aircraft, an absence from the network will result in only one report being missed from each other mobile; (b) a station is normally marked as unreachable if a number of reports have been missed from it – therefore if typically only one report from this station has been missed by other stations in 10 seconds, the unreachable status of this station, as registered by other stations, will not generally be triggered (assuming that for other stations L1 and TL3 are set at default values of 3 and 5 seconds respectively); (c) this station may have a transmission requirement to meet for different applications – for many applications an inability to transmit for 10 seconds will not represent a significant degradation of continuity or integrity.

Requirement reference	
	Network entry using plea/response procedures
	Plea transmission procedures
5.4.4.3.4h	A station desiring to transmit synchronization bursts on a VDL Mode 4 channel, which has been tuned to the channel for at least TL6 slots but which is prevented from transmission by lack of a valid reservation table, shall identify a set of peer stations S with the highest reported altitudes.
5.4.4.3.4i	The size of S, S_count, shall be defined by S_count = min((number of viable peer stations identified), CG1_plea).
5.4.4.3.4j	A peer station shall be considered viable if it is a mobile station which is known to have transmitted a synchronization burst on the channel on the normal slot boundary with the a/d bit set = 0, or if it is a ground station which is known to have transmitted a synchronization burst on the channel.
5.4.4.3.4k	If S_count is equal to zero, the TL6 timer shall be reset.
5.4.4.3.4l	If S_count is greater than 0, the station shall transmit a plea to a peer station selected randomly from S.
5.4.4.3.4m	The plea shall be transmitted using the response reservation encoding in clause 5.2.18.1 (the destination address bits being set to the address of the selected peer station), with transmission starting on the delayed burst slot boundary of an unoccupied slot.
5.4.4.3.4n	The station shall continue to transmit plea requests, using the Q5 parameters specified in table 5.73, until one of the following conditions is satisfied: a) it receives a plea response directed reservation request; b) it receives an autotune directed reservation request; c) it creates the necessary number of synchronization burst streams for the channel.
5.4.4.3.4o	On each transmission of a plea request, the set S shall be rebuilt and the destination shall be selected randomly from the set.
5.4.4.3.4p	A network entry burst (defined in table 5.3) shall have length n≤11 octets excluding burst flags and bit stuffing.
5.4.4.3.4q	And shall have a priority of 14.
5.4.4.3.4r	The information field, if present, shall be set to 0 on transmit and shall be ignored on receive.
	Plea response transmission procedures
5.4.4.3.5	Upon receiving a network entry burst with a response reservation addressed to itself (i.e. a plea), a station shall take the following actions.
5.4.4.3.6	If the station has observed the given frequency for at least the previous 60 s, and has not initiated a network entry or re-entry procedure within the previous 60 s, it shall transmit a plea response burst as defined in clauses 5.2.16.1.10 to 5.2.16.1.14 containing min (12, number of reservations required to allow one minute of transmissions at the default sync burst rate for this channel) reservations or else if the transmission rate is not known once per 10 s reservations.
5.4.4.3.7	These reservations shall be identified as follows: 1) unexpired reservations from any prior plea response addressed to the requesting station; 2) any periodic reservations for the requesting station, not otherwise contained in a prior plea response and which a) do not conflict with other known reservations, and b) can be appended to a possible list of reservations in accordance with item (1) above (considering the encoding constraints of the plea response); 3) additional reservations as required, using the selection parameters of table 5.73.
NOTE 1: A station may consider its reservation table for a channel invalid, for the purpose of selecting slots for synchronization bursts, if it has been tuned to a channel for less than M1 slots or if the conditions for rapid network entry (as defined in table 5.72b) are satisfied. NOTE 2: If the station has less than nominal update rate (nr) reservations for synchronization bursts over the next M1 slots at the time that the last directed reservation is executed, it may repeat the plea transmission procedure in order to generate additional reservations. The TL6 timer will not be running and is not required in this scenario. NOTE 3: In the event that a station issues multiple pleas and receives multiple responses, the final plea/response interaction could result in a total number of synchronization burst streams per M1 slots which exceeds the nominal update rate (nr). In this case, the station would be required to cancel some of the reservations and shift individual nominal slots (or sets of nominal slots) in order to form a periodic sequence of synchronization bursts in time in accordance with clause 5.2.10.5.1.	

Table 5.73: Plea response parameters

Symbol	Parameter name	Default
Q1	Priority	14
Q2a	Slot selection range constraint for level 1	150 NM
Q2b	Slot selection range constraint for level 2	150 NM
Q2c	Slot selection range constraint for level 3	0 NM
Q2d	Slot selection range constraint for level 4	300 NM
Q3	Replace queued data	TRUE
Q4	Number of available slots	3

Requirement reference	
5.4.4.3.8	Otherwise the station shall ignore the burst.
5.4.4.3.9	If the station cannot transmit the plea response within TL5 seconds of receiving the plea, the plea response shall be purged and not transmitted.
	Recommendation
5.4.4.3.10	The station should attempt to transmit the plea response as soon as possible following the plea (while still selecting the transmit slot randomly).
5.4.4.3.11	The first reservation contained in the plea response should occur as soon as possible in time following the plea response, but not sooner than V52 slots.
5.4.4.3.12	The station should attempt to reserve slots which are currently unreserved, and which are related to available slots for which the plea recipient can transmit BND reservations.
NOTE: The constraints on transmission time prevent possible race conditions and conflicting reservation sets associated with multiple exchanges of pleas and plea responses. The requirement to include unexpired reservations, if they are known, minimizes the likelihood of wasted channel resources in the case where a station fails to hear its requested plea response, but where other station(s) in the airspace heard it and allocated the appropriate slots.	

Requirement reference	
	Plea response reception procedures
5.4.4.3.12a	A station receiving a plea response shall transmit in the reserved slots. (See note 1).
	Network entry via BND
5.4.4.3.12b	A station intending to initiate synchronization burst transmissions on a channel, which has insufficient reservations for its intended number of synchronization burst streams, and which cannot make periodic reservations, shall transmit a series of delayed network entry bursts with BND reservations. (See note 2).
5.4.4.3.12c	The number per minute of delayed network entry bursts with BND reservations shall be equal to or less than the intended number of synchronization burst streams per minute.
5.4.4.3.12d	The station shall stop transmitting delayed network entry bursts with a BND reservation if it has set up the required number of streams or it receives a plea response or a directed reservation request.
5.4.4.3.12e	If it receives a plea response or directed reservation request, and thus does not require the reservations created by previous BND transmissions, it shall include a no reservation field in the synchronization bursts transmitted in the unnecessary slots.
5.4.4.3.12f	At most one delayed network entry burst using a BND broadcast reservation shall be made per intended stream.
5.4.4.3.12g	A station which has reservations for synchronization burst transmissions and which intends to continue operations on the channel, but which is unable to make periodic reservations, shall use BND reservations in lieu of periodic reservations.
5.4.4.3.12h	A network entry burst (defined in table 5.3) shall have length n ≤ 11 octets excluding burst flags and bit stuffing and shall have a priority of 14.
5.4.4.3.12i	The information field, if present, shall be set to 0 on transmit and shall be ignored on receive.
	Network entry by full-slot random transmission
5.4.4.3.13	In the event that a station has listened to a channel for M1 + 126 slots prior to net entry, a station shall use one of two procedures to begin transmitting synchronization bursts: a) use default random transmission protocols with combined periodic/incremental reservation types to place each new periodic reservation and to simultaneously reserve the next selected slot in the same superframe for the transmission containing the next periodic reservation; or b) use default random transmission protocols to transmit a plea response addressed to itself, followed by synchronization bursts with periodic reservations in the reserved slots. NOTE: These may be affected by other reservations detected after transmission of the plea response.
NOTE 1: If the station has not yet declared/transmitted an existing set of periodic broadcast reservations for all required synchronization bursts on a given frequency, it may use the reservation fields in the slots reserved by the plea response to set up these periodic streams by using periodic broadcast reservations (if it has been listening to the channel for at least 60 seconds) or BND reservations (if it has been listening to the channel long enough to use the BND reservation, but not long enough for a periodic reservation). A station may also transmit a null reservation (see clause 5.2.9) or a periodic broadcast cancellation (see clause 5.2.10.5.9) in the slots reserved by the plea response. See clause 5.2.10.5.1 for requirements regarding transmissions after the first M1 slots. NOTE 2: A station may be temporarily unable to make periodic broadcast reservations if its existing reservations are due to a recently received plea response or if it determines that it has become "exposed" (i.e. CG1 exceeded CG1_limit within the previous 60 seconds). NOTE 3: These default procedures may also be used when a station reverts to autonomous operations after a period of directed operations on the GSC(s). Consequently, the station will have a valid reservation table at the time of its reversion to autonomous mode.	

5.4.5 Additional material for ADS-B applications

5.4.5.1 Information field formats

Requirement reference	
5.4.5.1.1	The information field formats (defined in the present document) shall be as defined in table 5.74.

Table 5.74: ADS-B information fields

Information field ID (id)	ID extension 1 (id1)	ID extension 2 (id2)	Information field name
B to E hex	Not present	not present	Available for future use
F hex	Not present	not present	No information field provided

5.4.5.2 ADS-B request format

Requirement reference	
5.4.5.2.1	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, and include the information as shown in table 5.75.
5.4.5.2.2	The information subfields shall be encoded according to table 5.76.

Table 5.75: ADS-B request bit encoding

Description	Octet	Bit number							
		8	7	6	5	4	3	2	1
requested base altitude (r-b/a), r-mi_1 (bit 8 = 0), burst ID	5	0	0	r-b/a_2	r-b/a_1	0	0	0	1

Table 5.76: ADS-B request field encoding

Subfield	Range	Encoding	Out of Scale	Notes
Requested base altitude (r-b/a)	0 to 3	0 = report either barometric or geometric 1 = report barometric or, if not available, report geometric 2 = report geometric or, if not available, report barometric 3 = reserved for future use		

5.5 Void

5.6 Definitions for Compact Position Reporting

5.6.1 Introduction

5.6.2 Parameter symbols, data types, constants and variables

5.6.2.1 Parameter symbols

Table 5.77: Summary of parameter symbols for CPR

Parameter	Name	clause or table defined in
TR1	Maximum age for use in global decode	clauses 5.6.6.2.2 to 5.6.6.2.4
TR2	Maximum time between global updates	clauses 5.6.6.2.5 to 5.6.6.2.6

5.6.2.2 Data types

Requirement reference	
5.6.2.2.1	All calculations in this clause shall use signed integers.
5.6.2.2.2	Results of calculations to perform encoding and decoding shall match the results when performed with 64-bit signed integer operations.
NOTE: Division is therefore considered to be integer division (the result of a division is truncated to an integer).	

5.6.2.3 Constants

Requirement reference	
5.6.2.3.1	Constants used in the description of CPR shall have the values defined in table 5.78.

Table 5.78: Constants used in CPR calculations

Type	Name	Value	Description
Integer	LAT_z	9	Number of zones from 0° to 90° latitude
Integer	MAX_c	2^{51}	Maximum value for longitude and latitude
Integer	MAX_T^{lat}	2^{12}-1	Maximum transmitted latitude value
Integer	MAX_T^{lon}	2^{14}-1	Maximum transmitted longitude value

5.6.2.4 Variables

Requirement reference	
5.6.2.4.1	Variables used in CPR calculations shall have the type and range restrictions defined in table 5.79.
5.6.2.4.2	[A,B] shall mean greater than or equal to A and less than or equal to B.
5.6.2.4.3	[A,B) shall mean greater than or equal to A and less than B.
NOTE 1: CPR calculations convert between three representations of a station's position: a) The external representation of latitude and longitude which is meaningful to applications. b) The internal representation which provides an integer representation of the position. c) The link representation which is the encoded position. Tables 5.79 and 5.80 divide the variables and functions used in the CPR calculations into these three categories. NOTE 2: All CPR computations use integer-valued longitude and latitude (in the range of $[0, MAX_C]$, where each step is $\frac{360}{MAX_C + 1} \approx 0.1598721155 \times 10^{-12}$ degrees). The conversion from arbitrary-precision real numbers is detailed in the 4.2.5 to 4.2.6.	

Table 5.79: Variables used in CPR calculations

Type	Name	Range	Description
External representation			
Real	Latitude	[0,90], [270, 360)	The input latitude Note that a latitude of [-90,0] maps to [270,360].
Real	Longitude	[0,360)	The input longitude.
Internal representation			
Integer	type, $type_{last}$	0 or 1	The type of CPR (0 = even, 1 = odd).
Integer	$clat_{in}$, $clon_{in}$	[0, Max_c]	Latitude and longitude to be encoded.
Integer	tmp_n	[0, Max_c]	Temporary variable number n. Only used to make expressions and functions more readable.
Integer	$clat_{ref}$, $clon_{ref}$	[0, Max_c]	Reference latitude and longitude for local decoding.
Integer	X		Any integer.
Integer	pos_1, pos_2	[0, Max_c]	A latitude or longitude.
Integer	$clat_{dec}$, $clon_{dec}$	[0, Max_c]	Decoded latitude and longitude.
Integer	Bits	3,5 or 7	Number of bits for the magnitude offset.
Integer	lat_{offs}, lon_{offs}	[0,2^{bits}-1]	Latitude and Longitude offset.
Integer	s_{lat}, s_{lon}	0 or 1	Sign of the latitude and longitude offset.
Integer	lat_p	[0, 18]	The latitude patch.
Integer	lon_p	[0, 35]	The longitude patch.
Link representation			
Integer	Cprf	0 or 1	CPR format even/odd
Integer	Lat	$\left[0, MAX_T^{lat}\right]$	Encoded latitude
Integer	Lon	$\left[0, MAX_T^{lat}\right]$	Encoded longitude
Integer	lat_{ref}, lat_0, lat_1	$\left[0, MAX_T^{lat}\right]$	Encoded latitude
Integer	lon_{ref}, lon_0, lon_1	$\left[0, MAX_T^{lat}\right]$	Encoded longitude
Integer	lat4, lat6, lat8	[0, $2^{bits + 1}$ -1]	Encoded latitude offset
Integer	lon4, lon6, lon8	[0, $2^{bits + 1}$ -1]	Encoded longitude offset
Integer	Pid	[0, 179]	Encoded patch ID

5.6.2.5 Functions

Requirement reference	
5.6.2.5.1	Functions used in CPR shall have the input parameters and return values defined in table 5.80.

Table 5.80: Input parameters and return values for functions used in CPR calculations

Type	Name	Description
Function returns value in internal representation		
Integer	nz (type)	Number of zones depending on the type (odd/even) of CPR format.
Integer	dlat (type)	Latitude patch size for type.
Integer	nl ($clat_{dec}$, type)	Looks up the value in the transition level table 5.81.
Integer	dlon ($clat_{dec}$, type)	Longitude patch size at latitude clatdec for type typ5.
Integer	lat_{offs}(lat, lat_{ref})	Latitude zone offset
Integer	lon_{offs}(lon, lon_{ref})	Longitude zone offset
Integer	dec_{lat} ($clat_{ref}$, lat, lat_{ref}, type)	Local latitude decoding.
Integer	dec_{lon} ($clat_{dec}$, $clon_{ref}$, lon, lon_{ref}, type)	Local longitude decoding.
Integer	lat_{seg} (lat_0, lat_1, $type_{last}$)	Latitude segment for global decoding.
Integer	lon_{seg} (lon_0, lon_1, $clat_{dec}$, $type_{last}$)	Longitude segment for global decoding.
Integer	$globalDec_{lat}$ (lat_0, lat_1, $type_{last}$)	Global latitude global.
Integer	$globalDec_{lon}$ (lon_0, lon_1, $clat_{dec}$, $type_{last}$)	Global longitude global.
Integer	fix (x)	Converts negative co-ordinates to positive.
Integer	lookup ($clat_{in}$, type)	The value that corresponds to $clat_{in}$ and type in the transition level table.
Integer	diff (pos_1, pos_2)	The (shortest) distance between pos_1 and pos_2.
Integer	sign (pos_1, pos_2)	The sign of diff (pos_1, pos_2).
Integer	$offset_{dec}^{lat}(lat_{offs}, s_{lat}, bits, type)$	Calculates the true offset for the latitude offset given in bits.
Integer	$offset_{dec}^{lon}(clat_{dec}, lon_{offs}, s_{lon}, bits, type)$	Calculates the true offset for the longitude offset given in bits.
Integer	$fullDec_{lat}$ (lat, lat_p, type)	Decodes full position latitude.
Integer	$fullDec_{lon}$ ($clat_{dec}$, lon, lon_p, type)	Decodes full position longitude.
Function returns value in link representation		
Integer	enc_{lat} ($clat_{in}$, type)	Returns the CPR encoded value for $clat_{in}$ using type.
Integer	enc_{lon} ($clat_{dec}$, $clon_{in,}$ type)	Returns the CPR encoded value for $clon_{in}$ using type.
Integer	$offset_{enc}^{lat}(clat_{in}, clat_{dec}, bits, type)$	The difference between $clat_{in}$ and $clat_{dec}$ expressed using bits.
Integer	$offset_{enc}^{lon}(clat_{dec}, clon_{in}, clon_{dec}, bits, type)$	The difference between lon_{in} and $clon_{dec}$ expressed using bits.
Integer	enc_{patch} (lat_p, lon_p)	Encode the patch id.

5.6.2.6 Patch constants

Requirement reference	
	Transition table
5.6.2.6.1	The function *lookup(clatin, type)* shall return the value in the number of zones (even or odd, depending on *type*) column in table 5.81 for which the *clatin* value satisfies the restriction in the Range(integer) column.

Table 5.81: Transition table for *lookup* function

Range (degrees)	Range (integer)	Number of zones Even	Number of zones Odd
<13,518674176405572	<84559299976949	35	34
[13.518674176405572,19.162797152134097)	[84559299976949,119863286269066)	34	33
[19.162797152134097,23.5247169626056)	[119863286269066,147147092426093)	33	32
[23.5247169626056,27.228512609375226)	[147147092426093,170314332279771)	32	31
[27.228512609375226,30.51543280332421)	[170314332279771,190874016391806)	31	30
[30.51543280332421,33.50899730287358)	[190874016391806,209598760787195)	30	29
[33.50899730287358,36.28248037044658)	[209598760787195,226946895939473)	29	28
[36.28248037044658,38.883571527761575)	[226946895939473,243216719782307)	28	27
[38.883571527761575,41.34536944123708)	[243216719782307,258615264457015)	27	26
[41.34536944123708,43.691961273699334)	[258615264457015,273293195154609)	26	25
[43.691961273699334,45.941527811563425)	[273293195154609,287364232684706)	25	24
[45.941527811563425,48.10819571981785)	[287364232684706,300916739329498)	24	23
[48.10819571981785,50.20320392571675)	[300916739329498,314021014573143)	23	22
[50.20320392571675,52.23567067731592)	[314021014573143,326734093052511)	22	21
[52.23567067731592,54.213116139057256)	[326734093052511,339103013392294)	21	20
[54.213116139057256,56.14182888275907)	[339103013392294,351167110605961)	20	19
[56.14182888275907,58.02712896497076)	[351167110605961,362959661644475)	19	18
[58.02712896497076,59.87356014060077)	[362959661644475,374509087692437)	18	17
[59.87356014060077,61.68503184003544)	[374509087692437,385839842234890)	17	16
[61.68503184003544,63.46492412462716)	[385839842234890,396973067553844)	16	15
[63.46492412462716,65.2161639281094)	[396973067553844,407927071618287)	15	14
[65.2161639281094,66.9412773021877)	[407927071618287,418717654880330)	14	13
[66.9412773021877,68.6424192797632)	[418717654880330,429358297069654)	13	12
[68.6424192797632,70.32137954962614)	[429358297069654,439860192688716)	12	11
[70.32137954962614,71.97955727480327)	[439860192688716,450232093501524)	11	10
[71.97955727480327,73.61788995824008)	[450232093501524,460479863588517)	10	9
[73.61788995824008,75.23670452702919)	[460479863588517,470605547878490)	9	8
[75.23670452702919,76.83542194177753)	[470605547878490,480605524480339)	8	7
[76.83542194177753,78.41195676510516)	[480605524480339,490466748984332)	7	6
[78.41195676510516,79.9614066817654)	[490466748984332,500158557411138)	6	5
[79.9614066817654,81.47284075679195)	[500158557411138,509612576768200)	5	4
[81.47284075679195,82.91989876526003)	[509612576768200,518663923862256)	4	3
[82.91989876526003,84.22404437738102)	[518663923862256,526821353991124)	3	2
[84.22404437738102,84.99999999999986)	[526821353991124,531674956009016)	2	1
≥ 84.99999999999986	≥ 531674956009016	1	1
NOTE: The table is symmetrical for latitudes in the range of $\left[3\cdot\lfloor MAX_C/4\rfloor, MAX_C\right]$ (i.e. [–90, 0] degrees). This means that for latitude, the range of 0 to 90 degrees is mapped to [$0-\lceil MAX_C/4\rceil$] and [−90 – 0 = 270 – 360] is mapped to [$3\cdot\lfloor MAX_C/4\rfloor - MAX_C$].			

Requirement reference	
	Patch size functions
5.6.2.6.2	The size of a latitude and longitude patch shall be: $nz(type) = 4\cdot LAT_Z - type$ $dlat(type) = \frac{MAX_C}{nz(type)}$ $nl(clat_{in}, type) = \begin{cases} lookup(clat_{in}, type) & \text{if } clat_{in} < MAX_C/2 \\ lookup(MAX_C - clat_{in}, type) & \text{else} \end{cases}$ $dlon(clat_{in}, type) = \frac{MAX_C}{nl(clat_{in}, type)}$

5.6.3 Fixed data field position encoding

5.6.3.1 General

Requirement reference	
5.6.3.1.1	Given an arbitrary position *latitude* and *longitude* and a desired *type* (odd or even), the *lat, lon* and *cprf* sub-fields in the fixed data field of the synchronization burst shall be set to the value of $enc_{lat}()$ and $enc_{lon}()$ computed as defined in clauses 5.6.3.2.1 to 5.6.3.3.2.

5.6.3.2 Input parameters

Requirement reference	
5.6.3.2.1	The input parameters used for fixed data field encoding shall be defined as follows: latitude = latitude to be encoded. longitude = longitude to be encoded. type = type of encoding (odd or even).

5.6.3.3 Calculations

Requirement reference	
	Latitude
5.6.3.3.1	The encoded fixed latitude component shall be calculated as follows: $clat_{in} = \frac{(latitude)\cdot(MAX_C + 1)}{360}$ $lat = enc_{lat}(clat_{in}, type) = \frac{\left(nz(type)\cdot \mathrm{mod}(clat_{in}, dlat(type)) + \frac{MAX_C}{2\cdot MAX_T^{lat}}\right)}{\left(\frac{MAX_C}{MAX_T^{lat}}\right)}$
	Longitude
5.6.3.3.2	The encoded fixed longitude component shall be calculated as follows: $clon_{in} = \frac{(longitude)\cdot(MAX_C + 1)}{360}$ $lon = enc_{lon}(clat_{dec}, clon_{in}, type) = \frac{\left(nl(clat_{dec}, type)\cdot \mathrm{mod}(clon_{in}, dlon(clat_{dec}, type)) + \frac{MAX_C}{2\cdot MAX_T^{lon}}\right)}{\left(\frac{MAX_C}{MAX_T^{lon}}\right)}$

5.6.4 Fixed data field position local decoding

5.6.4.1 General

Requirement reference	
5.6.4.1.1	When the position report processing state machine (see clause 5.6.6.3.6) indicates that local decoding is to be performed, then the fixed data field position shall be decoded using a single position report and an unambiguous global reference location.
5.6.4.1.2	The calculation shall return the latitude, longitude and type sub-fields.
NOTE: Local decoding uses a single report from the target and a reference location which can be either the position of the receiver (if only one type of report has been received from a target) or the last decoded position for the target (if the unambiguous global position of the target is known).	

5.6.4.2 Input parameters

Requirement reference	
5.6.4.2.1	The input parameters used for fixed data field decoding shall be defined as follows: $clat_{ref}$ = reference latitude. $clon_{ref}$ = reference longitude. *lat* = CPR encoded latitude to be decoded. *lon* = CPR encoded longitude to be decoded. *cprf* = CPR format even/odd.

5.6.4.3 Calculations

Requirement reference	
	Supporting Function
5.6.4.3.1	The supporting function for calculating the decoded fixed position field shall be as follows: $fix(x)=\begin{cases} x+1+MAX_c & \text{if } x<0 \\ x & \text{else} \end{cases}$
	Latitude
5.6.4.3.2	The decoded fixed latitude component shall be calculated as follows: $type = cprf$ $lat_{ref} = enc_{lat}(clat_{ref}, type)$ $lat_{offs}(lat, lat_{ref}) = \begin{cases} 1 & \text{if } (lat_{ref} - lat) > \frac{MAX_T^{lat}}{2} \\ -1 & \text{if } (lat_{ref} - lat) < -\frac{MAX_T^{lat}}{2} \\ 0 & \text{else} \end{cases}$ $tmp_1 = dlat(type) \cdot \left(\frac{clat_{ref}}{dlat(type)} + lat_{offs}(lat, lat_{ref}) \right)$

Requirement reference	
	$$clat_{dec} = dec_{lat}\left(clat_{ref}, lat, lat_{ref}, type\right) = fix\left(\frac{\left(\left(\frac{MAX_C}{MAX_T^{lat}}\right)\cdot lat\right)}{nz(type)} + tmp_1\right)$$ $$latitude = \frac{\left(clat_{dec} + offset_{dec}^{lat}\left(lat_{offs}, s_{lat}, bits, type\right)\right)\cdot 360}{\left(MAX_C + 1\right)}$$
	Longitude
5.6.4.3.3	The decoded fixed longitude component shall be calculated as follows: $$lon_{ref} = enc_{lon}\left(clat_{dec}, clon_{ref}, type\right)$$ $$lon_{offs}\left(lon, lon_{ref}\right) = \begin{cases} 1 \text{ if } \left(lon_{ref} - lon\right) > \frac{MAX_T^{lon}}{2} \\ -1 \text{ if } \left(lon_{ref} - lon\right) < -\frac{MAX_T^{lon}}{2} \\ 0 \text{ else} \end{cases}$$ $$tmp_2 = dlon\left(clat_{dec}, type\right)\cdot\left(\frac{clon_{ref}}{dlon\left(clat_{dec}, type\right)} + lon_{offs}\left(lon, lon_{ref}\right)\right)$$ $$clon_{dec} = dec_{lon}\left(clat_{dec}, clon_{ref}, lon, lon_{ref}, type\right) = fix\left(\frac{\left(\left(\frac{MAX_C}{MAX_T^{lon}}\right)\cdot lon\right)}{nl\left(clat_{dec}, type\right)} + tmp_2\right)$$ $$longitude = \frac{\left(clon_{dec} + offset_{dec}^{lon}\left(clat_{dec}, lon_{offs}, s_{lon}, bits, type\right)\right)\cdot 360}{\left(MAX_C + 1\right)}$$

5.6.5 Fixed data field position global decoding

5.6.5.1 General

Requirement reference	
5.6.5.1.1	When the position report processing state machine (see clause 5.6.6.3.6) indicates that global decoding is to be performed, then the fixed data field position shall be decoded using the most recently received odd and even fixed data field positions.
5.6.5.1.2	The calculation shall return the *latitude*, *longitude* and *type* fields.
NOTE: The global decoding is guaranteed to succeed if the target has travelled less than 8.4 NM between the odd and even position report and if the target has not crossed any transition latitude between the reports.	

5.6.5.2 Input parameters

Requirement reference	
5.6.5.2.1	The input parameters used for fixed data field global decoding shall be defined as follows: lat_0 = even CPR encoded latitude to be decoded. lon_0 = even CPR encoded longitude to be decoded. lat_1 = odd CPR encoded latitude to be decoded. lon_1 = odd CPR encoded longitude to be decoded. *cprf* = type of encoding (odd or even) for the most recent of the two CPR reports.

5.6.5.3 Transition level straddling

Requirement reference	
5.6.5.3.1	If, $nl(globalDec_{lat}(lat_0, lat_1, 1, 0) \neq nl(globalDec_{lat}(lat_0, lat_1, 0),0)$ then decoding as defined in clause 4.10.3.3 shall be computed instead of a global decode.
NOTE: This situation occurs when the target has straddled a transition latitude.	

5.6.5.4 Calculations

Requirement reference	
	Latitude
5.6.5.4.1	The globally decoded fixed latitude component shall be calculated as follows: $type_{last} = cprf$ $tmp_3 = \frac{\left(lat_0 \cdot nz(1) + 2 \cdot nz(type_{last}) \cdot MAX_T^{lat} + \frac{MAX_T^{lat}}{2} - lat_1 \cdot nz(0)\right)}{MAX_T^{lat}}$ $lat_{seg}(lat_0, lat_1, type_{last}) = \mathrm{mod}(tmp_3, nz(type_{last}))$ $tmp_4 = lat_{seg}(lat_0, lat_1, type_{last}) \cdot dlat(type_{last})$ $clat_{dec} = globalDec_{lat}(lat_0, lat_1, type_{last}) = tmp_4 + \frac{\left(\left(\frac{MAX_C}{MAX_T^{lat}}\right) \cdot lat_{type_{last}}\right)}{nz(type_{last})}$

Requirement reference	
	Longitude
5.6.5.4.2	The globally decoded fixed longitude component shall be calculated as follows: $$tmp_5 = \frac{\left(lon_0 \cdot nl(clat_{dec}, 1) + 2 \cdot nl(clat_{dec}, type_{last}) \cdot MAX_T^{lon} + \frac{MAX_T^{lon}}{2} - lon_1 \cdot nl(clat_{dec}, 0) \right)}{MAX_T^{lon}}$$ $$lon_{seg}(lon_0, lon_1, clat_{dec}, type_{last}) = \mathrm{mod}(tmp_5, nl(clat_{dec}, type_{last}))$$ $$tmp_6 = lon_{seg}(lon_0, lon_1, clat_{dec}, type_{last}) \cdot dlon(clat_{dec}, type_{last})$$ $$clon_{dec} = globalDec_{lon}(lon_0, lon_1, clat_{dec}, type_{last}) = tmp_6 + \frac{\left(\left(\frac{MAX_C}{MAX_T^{lon}} \right) \cdot lon_{type_{last}} \right)}{nl(clat_{dec}, type_{last})}$$

5.6.6 Position Report Processing

5.6.6.1 Services

Requirement reference	
5.6.6.1.1	The PECT (see clause 5.4.4.2) shall maintain sufficient history of received targets to enable unambiguous global position to be determined.

5.6.6.2 Position report parameters

Requirement reference	
5.6.6.2.1	The position report parameters shall be as defined in table 5.82.

Table 5.82: Position report processing parameters

Symbol	Parameter name	Minimum	Maximum	Default	Increment
TR1	Maximum age for use in global decode	1 s	60 s	30 s	1 s
TR2	Maximum time between global updates	1 s	240 s	60 s	1 s

Requirement reference	
	Parameter TR1 (maximum age for use in global decode)
5.6.6.2.2	The parameter TR1 shall be the maximum age of a report for its use in a global decode.
5.6.6.2.3	The timer shall be started (or restarted) as defined in table 5.83.
5.6.6.2.4	If it expires the report shall not be valid for use in a global decode.
	Parameter TR2 (maximum time between global updates)
5.6.6.2.5	The parameter TR2 shall be the maximum time between global updates.
5.6.6.2.6	The timer shall be started (or restarted) as defined in table 5.83.

5.6.6.3 Position report processing procedures

Requirement reference	
	Position report processing state machine
5.6.6.3.1	For each station maintained in the PECT (see clause 5.4.4.2), the station shall maintain the record of the last received position report and a position report processing state machine with the following states:
5.6.6.3.2	a) State 1 shall indicate that no position report has been received and represents the initial state of the position report processing state machine;
5.6.6.3.3	b) State 2 shall indicate that a position report has been received but that no position has been decoded;
5.6.6.3.4	c) State 3 shall indicate that a position report has been received and that a local position has been decoded;
5.6.6.3.5	d) State 4 shall indicate that a position report has been received and that a global position has been decoded.
	Position report processing state machine transitions
5.6.6.3.6	On receipt of a position report, the station shall update its state machine as defined in table 5.83 and report target position quality to the application.
5.6.6.3.7	If the target becomes unreachable, due to expiration of L1, the state machine for that target shall be set to State 1, whatever the previous state machine level for that target.

Table 5.83: State transitions for position report processing

<table>
<tr><td colspan="4">In State</td><td>1</td><td colspan="2">2</td><td colspan="2">3</td><td colspan="2">4</td></tr>
<tr><td colspan="4">Last report</td><td>None</td><td>Even</td><td>Odd</td><td>Even</td><td>Odd</td><td>Even</td><td>Odd</td></tr>
<tr><td colspan="4">Target position quality</td><td>None</td><td>None</td><td>None</td><td>Local</td><td>Local</td><td>Global</td><td>Global</td></tr>
<tr><td rowspan="2">Received position report type</td><td rowspan="2">Own Posi-tion</td><td colspan="2">Timers (exp = expired)</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>TR1</td><td>TR2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Even or odd with patch ID</td><td>not appli-cable</td><td>not appli-cable</td><td>not appli-cable</td><td>N = 4
C = GL
resTR1
resTR2</td><td colspan="2">N = 4, C = GL
resTR1, resTR2</td><td colspan="2">N = 4, C = GL
resTR1, resTR2</td><td colspan="2">N = 4, C = GL
resTR1, resTR2</td></tr>
<tr><td rowspan="8">Even</td><td rowspan="4">Yes</td><td rowspan="2">Not exp</td><td>Not exp</td><td rowspan="4">N = 3
C = L1
ResTR1</td><td rowspan="4">N = 3
C = L1
resTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td><td rowspan="4">N = 3
C = L1
resTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td><td>N = 4
C = L2
resTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td></tr>
<tr><td>Exp</td><td>N = 3
C = L1
resTR1</td></tr>
<tr><td rowspan="2">Exp</td><td>Not exp</td><td rowspan="2">N = 3
C = L1
resTR1</td><td rowspan="2">N = 3
C = L1
resTR1</td><td colspan="2">N = 4, C = L2, resTR1</td></tr>
<tr><td>Exp</td><td colspan="2">N = 3, C = L1, resTR1</td></tr>
<tr><td rowspan="4">No</td><td rowspan="2">Not exp</td><td>Not exp</td><td rowspan="4">N = 2
C = NO
ResTR1</td><td rowspan="4">N = 2
C = NO
resTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td><td rowspan="4">N = 2
C = NO
resTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td><td>N = 4
C = L2
resTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td></tr>
<tr><td>Exp</td><td>N = 2
C = NO
resTR1</td></tr>
<tr><td rowspan="2">Exp</td><td>Not exp</td><td rowspan="2">N = 2
C = NO
resTR1</td><td rowspan="2">N = 2
C = NO
resTR1</td><td colspan="2">N = 4, C = L2, resTR1</td></tr>
<tr><td>Exp</td><td colspan="2">N = 2, C = NO, resTR1</td></tr>
</table>

<table>
<tr><th colspan="4">In State</th><th>1</th><th colspan="2">2</th><th colspan="2">3</th><th colspan="2">4</th></tr>
<tr><th colspan="4">Last report</th><th>None</th><th>Even</th><th>Odd</th><th>Even</th><th>Odd</th><th>Even</th><th>Odd</th></tr>
<tr><th colspan="4">Target position quality</th><th>None</th><th>None</th><th>None</th><th>Local</th><th>Local</th><th>Global</th><th>Global</th></tr>
<tr><td rowspan="8">Odd</td><td rowspan="4">Yes</td><td rowspan="2">Not exp</td><td>Not exp</td><td rowspan="4">N = 3
C = L1
ResTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td><td rowspan="4">N = 3
C = L1
resTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td><td rowspan="4">N = 3
C = L1
resTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td><td>N = 4
C = L2
resTR1</td></tr>
<tr><td>Exp</td><td>N = 3
C = L1
resTR1</td></tr>
<tr><td rowspan="2">Exp</td><td>Not exp</td><td rowspan="2">N = 3
C = L1
resTR1</td><td rowspan="2">N = 3
C = L1
resTR1</td><td colspan="2">N = 4, C = L2, resTR1</td></tr>
<tr><td>Exp</td><td colspan="2">N = 3, C = L1, resTR1</td></tr>
<tr><td rowspan="4">No</td><td rowspan="2">Not exp</td><td>Not exp</td><td rowspan="4">N = 2
C = NO
ResTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td><td rowspan="4">N = 2
C = NO
resTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td><td rowspan="4">N = 2
C = NO
resTR1</td><td rowspan="2">N = 4
C = GL
resTR1
resTR2</td><td>N = 4
C = L2
resTR1</td></tr>
<tr><td>Exp</td><td>N = 2
C = NO
resTR1</td></tr>
<tr><td rowspan="2">Exp</td><td>Not exp</td><td rowspan="2">N = 2
C = NO
resTR1</td><td rowspan="2">N = 2
C = NO
resTR1</td><td colspan="2">N = 4, C = L2, resTR1</td></tr>
<tr><td>Exp</td><td colspan="2">N = 2, C = NO, resTR1</td></tr>
<tr><td colspan="11">NOTE 1: State transitions depend on:
a) the initial state;
b) the type (even or odd) of the last received position report;
c) the type of received position report (even/odd/with patch ID/without patch ID);
d) whether or not the station has knowledge of its own position; and
e) the state of timers TR1 and TR2.
NOTE 2: The station carries out the actions defined as "C =" in table 5.83:
a) C = GL: carry out global decoding as defined in clause 5.6.5 if there is no patch ID, or as defined in EN 302 842-3 [9] if there is a patch ID;
b) C = L1: carry out local decoding using station's own position as reference, as defined in clause 5.6.4;
c) C = L2: carry out local decoding using the last known position of the transmitting station as reference, as defined in clause 5.6.4;
d) C = NO: no decoding is carried out.
NOTE 3: The station restarts timers TR1 and TR2 as indicated in table 5.83:
a) resTR1: restart timer TR1
b) resTR2: restart timer TR2
NOTE 4: The state machine enters the state indicated as "N =" in table 5.83 and reports the target position quality to the application:
a) if the final state is 1 or 2, the station indicates a target position quality of "none";
b) if the final state is 3, the station indicates a target position quality of "local"; and
c) if the final state is 4, the station indicates a target position quality of "global".</td></tr>
</table>

Requirement reference	
	State machine transitions for transition level straddling
5.6.6.3.8	When a position report is received from a station which fulfils: a) the conditions for the initial state, last received report type, received report type, own position and timer states defined in table 5.84; b) the conditions defined in clause 5.6.5.3 are met (because the station has crossed a transition latitude), the receiving station shall update its state machine as defined in table 5.84 and report the target position quality to the application.
5.6.6.3.9	Otherwise the station shall process the report as defined in clause 5.6.6.3.6.

Table 5.84: State transitions for position report processing (transition level straddling)

In State				**1**	**2**		**3**		**4**	
Last report				**None**	**Even**	**Odd**	**Even**	**Odd**	**Even**	**Odd**
Target position quality				**None**	**None**	**None**	**Local**	**Local**	**Global**	**Global**
Received position report type	Own Posi-tion	Timers (exp = expired)								
		TR1	TR2							
Even	Yes	Not exp	Not exp	See table 5.83	See table 5.83	N = 3 C = L1 resTR1	See table 5.83	N = 3 C = L1 resTR1	See table 5.83	N = 4 C = L2 resTR1
			Exp							N =3 C = L1 res TR1
	No	Not exp	Not exp			N = 2 C = NO resTR1		N = 2 C = NO resTR1		N = 4 C = L2 resTR1
			Exp							N = 2 C = NO resTR1
Odd	Yes	Not exp	Not exp	See table 5.83	N = 3 C = L1 resTR1	See table 5.83	N = 3 C = L1 resTR1	See table 5.83	N = 4 C = L2 resTR1	See table 5.85
			Exp						N =3 C =L1 resTR1	
	No	Not exp	Not exp		N = 2 C = NO resTR1		N = 2 C = NO resTR1		N = 4 C = L2 resTR1	
			Exp						N = 2 C = NO resTR1	

6 General design requirements

6.1 Controls and indicators

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.1.

6.2 Warm up

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.3.

6.3 Airworthiness

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.4.

6.4 Intended function

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.5.

6.5 International Telecommunications Union regulations

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.6.

6.6 Fire protection

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.7.

6.7 Operation of controls

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.8.

6.8 Accessibility of controls

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.9.

6.9 Effects of tests

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.10.

6.10 Failure of the VDL Mode 4 equipment

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.11.

6.11 Software management

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.12.

6.12 Transceiver configuration

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.13.

6.13 Provision for multiple redundant VDL4 transceivers

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.14.

6.14 Reception capability

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.15.

6.15 Transmission capability

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.16.

6.16 Monitoring of proper operation

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.17.

6.17 Power-up self test

The equipment shall meet the requirements of EN 302 842-1 [4], clause 7.18.

7 Protocol test procedures

7.1 General

7.1.1 Input voltage

The equipment shall meet the requirements of EN 302 842-1 [4], clause 9.1.1.

7.1.2 Power input frequency

The equipment shall meet the requirements of EN 302 842-1 [4], clause 9.1.1.

7.1.3 Adjustment of equipment

The circuits of the equipment under test shall be properly aligned and otherwise adjusted in accordance with the manufacturer's recommended practices prior to application of the specified tests. Unless otherwise specified, no adjustments may be made once the test procedures have started.

7.1.4 Equipment configuration

Replacement or substitution of components or circuit modules within the equipment under test is not permitted once the test procedures have started.

The VDL Mode 4 mobile station shall undergo all testing with its operational software installed in the equipment. The software version number shall reflect the revision that is intended for approval.

The configuration data shall be set up so as to be representative of a real airborne installation. This configuration data set shall be completely documented. The configuration setup shall not be altered during the entire testing procedure.

7.1.5 Test equipment precautions

All equipment used in the performance of the tests should be identified by make, model and serial number where appropriate, and its latest calibration date. The specification of the accuracy of the test equipment is left to the calibration process prescribed by the agency which certifies the testing facility.

Precautions shall be taken during conduct of the tests to prevent the introduction of errors resulting from the improper connection of test instruments across the input and/or output impedances of the equipment under test.

If used to terminate the input or output of the equipment under test, the test instruments shall present the equivalent impedance to the equipment under test for which it was designed. Otherwise, the equipment under test shall be connected to loads having the impedance values for which it was designed.

7.1.6 Ambient conditions

Unless otherwise specified, all tests should be conducted under conditions of ambient room temperature, pressure and humidity, as defined in EN 302 842-1 [4], clause 9.1.1.

7.1.7 Connected loads

Unless otherwise specified, all tests shall be performed with the equipment connected to loads having the impedance values for which it is designed.

7.2 Required test rig

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

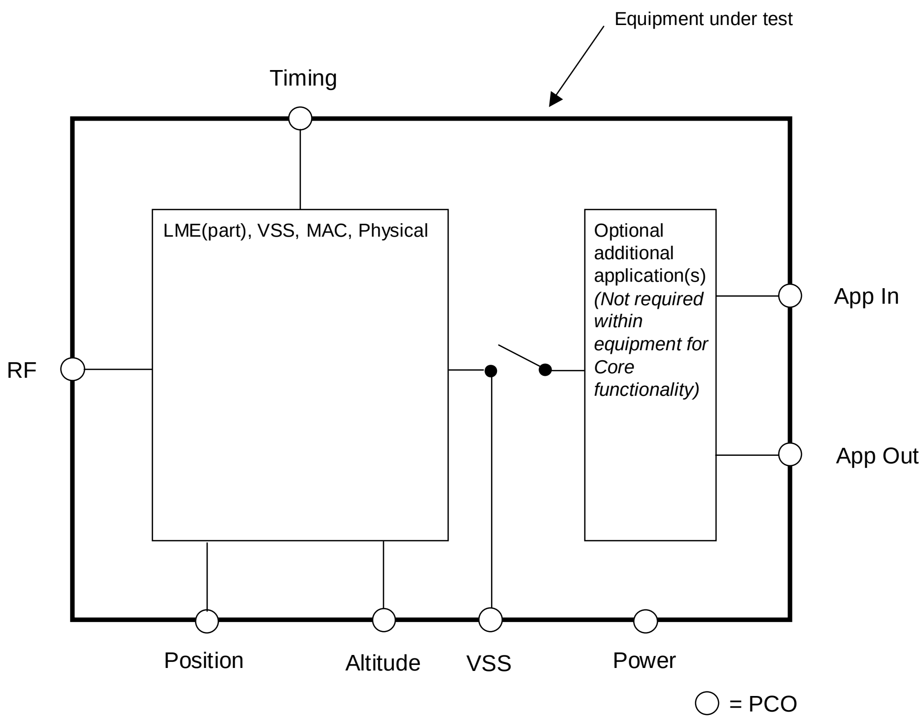

Figure 7.1: Location of PCOs

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

The PCOs identified in this figure are each associated with a test set which shall support the following:

RF:

- input to the equipment under test of a single burst or sequence of bursts, of specified content, one or more slots in length, commencing in a slot at a specified time, on a specified channel;
- recording of the time at which a burst containing specified content (per field) is output by the equipment under test, on any of three channels;
- simultaneous input to the equipment of bursts of specified content, commencing in a slot at a specified time, on two separate channels.

Timing:

- input of a reference time source compliant with the requirements of the present document;
- disabling of the input of reference time.

NOTE 1: Disabling of the timing source is required to force the equipment under test into secondary timing mode.

NOTE 2: In certain equipment architectures, the reference timing source may be incorporated internally within the equipment under test. Under such conditions, there is no requirement to expose the timing source itself, but a means must be provided to disable it as identified above.

Position:

- input to the equipment under test of a specified geographical position at a specified time;
- input to the equipment under test of position validity/quality to allow determination of position integrity (nic);
- disabling of the position source.

NOTE 3: Disabling of the position source is required to demonstrate that appropriate notification is provided by means of the Navigation Integrity Category (NIC) field.

NOTE 4: In certain equipment architectures, the position source may be incorporated internally into the equipment under test. Under such conditions, manufacturers will be required to perform alternative tests to those specified in the present document to demonstrate correct operation of the position encoding/decoding algorithms. In addition, a means must be provided to disable the position source as stated above.

Altitude:

- input to the equipment under test of a specified altitude at a specified time;
- disabling of the altitude source;
- configuration information identifying whether geometric or barometric altitude is provided.

NOTE 5: Disabling of the altitude source is required to demonstrate that appropriate notification is provided by means of the fixed synchronization burst.

NOTE 6: In certain equipment architectures, the altitude source may be incorporated internally into the equipment under test. Under such conditions, manufacturers will be required to perform alternative tests to those specified in the present document to demonstrate correct operation of the position encoding/decoding algorithms. In addition, a means must be provided to disable the position source as stated above.

VSS:

The VSS User PCO is not normally exposed during operational use of the VDL Mode 4 ADS-B system. It is available only during test mode, in which the internal application(s) are disconnected from the VSS and lower layers, as illustrated above.

The VSS User PCO is intended to provide a means to stimulate the VDL Mode 4 system independently of the internal applications, and to offer a mechanism to use test such features of the VSS sub-layer such as slot selection and reservation conflict processing which could not be tested adequately by any other means. At this PCO, functionality shall be provided to allow the User (i.e. test set) to:

- enable/disable autonomous synch bursts, and control of parameters TV11 min, TV11 max and V11 associated with their transmission;
- maintain a queue of random access transmissions, of user specified content, such that at least one burst is always in the transmit queue;
- establish a sequence of streams of periodic broadcasts, of user specified content, defined by parameters TV11 min, TV11 max, V11, V12, together with Quality of Service parameters Q2a to Q2d, Q4 and Q5;
- cancel an existing sequence of periodic streams;
- establish a sequence of incremental broadcasts, of user specified content, defined by parameters V21, V22, together with Quality of Service parameters Q2a to Q2d, Q4 and Q5;
- receive a notification that a non-zero version number has been detected;
- receive a notification in response to a request for transmission that no slot was available for selection.

AppIn:

Input to the equipment under test of any additional data required to support any internal applications. Tests for application functionality are outside the scope of the present document, and manufacturers are required to specify tests to demonstrate correct operation of any applications supported, including appropriate inputs via this PCO.

AppOut:

Output from the equipment under test of any data associated with internal application(s). Examples include ADS-B, TIS-B, FIS-B data for output to the crew. Tests for application functionality are outside the scope of the present document, and manufacturers are required to specify such tests to demonstrate correct operation of any applications supported, including appropriate outputs via this PCO.

NOTE 7: A display of ADS-B data built into the equipment may represent this PCO.

Power:

Power shall be applied at this PCO in accordance with clauses 7.1.1 and 7.1.2. The facility shall be provided to interrupt the power supply for a period between 150 ms and 15 s, upon an event being signalled from the surrounding test harness.

7.3 Protocol test-suite description methodology

The precise rules which control the functions of computer based equipment like the VDL Mode 4 mobile station, which involve sequential logic, require a rigorous interpretation which cannot always be readily achieved by plain text description. Therefore, a formal description has been used based on ISO/IEC 9646 [7]. The concepts of ISO/IEC 9646 [7] were, to maximum extent, applied to the VDL Mode 4 test procedures included in the present document. For convenience the underlying basic concepts are described in annex B.

7.4 Detailed protocol test procedures

The protocol 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.

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.

7.4.1 Test-suite overview

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

Table 7.1: Protocol test-suite overview

Test Case Name	Description
Physical_SysParams	To demonstrate that a station operates correctly at the limits of the physical layer system parameters.
Timing_Primary	To demonstrate that when primary timing is available, a transmission from the station complies with primary timing performance.
Timing_Secondary	To demonstrate that when primary timing is unavailable, a transmission from the station complies with secondary timing performance.
Timing_Secondary_Recover	To demonstrate that when primary timing becomes available to a station which is transmitting on secondary time, it reverts to using primary time.
CRC_Norm	To demonstrate that a station transmitting a burst will insert a valid CRC.
CRC_Rej	To demonstrate that a station receiving a burst with an invalid CRC will reject the burst.
Version_NonZero	To demonstrate that a station receiving a burst containing a non-zero version number will ignore the burst and inform the VSS user.
Queue_Replace	To demonstrate that a burst submitted to the VSS layer with Q3 set to TRUE will replace any queued data of the same type.
Queue_Norm	To demonstrate that a burst submitted to the VSS layer with Q3 set to FALSE will not replace any queued data of the same type.
MessageID_Invalid_A	To demonstrate that a unicast burst received with an invalid message ID will cause a General Failure burst to be transmitted.
MessageID_Invalid_B	To demonstrate that a burst with an invalid message ID not making a reservation for reply, causes no response to be made.
Reservation_Unrecognized	To demonstrate that an unrecognized reservation type will cause the packet to be rejected and an error logged.
Reservation_Invalid	To demonstrate that reception of a known reservation type with an invalid subfield causes the appropriate slots to be reserved, but not to transmit a response, nor pass the burst to a VSS user.
Reservation_Recognition	To demonstrate that a reservation will be recognized prior to the end of the slot following the transmission in which it was carried.
SlotSel_Level0_A	To demonstrate that a station will select a slot at level 0 when no slots are reserved.
SlotSel_Level0_B	To demonstrate that a station will select a slot at level 0, excluding those not meeting the criteria of any other level.
SlotSel_Level0_C	To demonstrate that a station will select a slot at level 0 in preference to those slots available at level 1.
SlotSel_Level0_D	To demonstrate that a station will select a slot at level 0 in preference to those slots available at level 2.
SlotSel_Level0_E	To demonstrate that a station will select a slot at level 0 in preference to those slots available at level 3.
SlotSel_Level0_F	To demonstrate that a station will select a slot at level 0 in preference to those slots available at level 4.
SlotSel_Level1_A	To demonstrate that a station will select a slot at level 1 when the appropriate criteria are satisfied.
SlotSel_Level1_B	To demonstrate that a station will select a slot at level 1, excluding those slots not meeting the criteria of level 1 or any lower priority level.
SlotSel_Level1_C	To demonstrate that a station will select a slot at level 1 in preference to those available at level 2.
SlotSel_Level1_D	To demonstrate that a station will select a slot at level 1 in preference to those available at level 3.
SlotSel_Level1_E	To demonstrate that a station will select a slot at level 1, in preference to those available at level 4.
SlotSel_Level1_F	To demonstrate that a station will select slots at level 1 from a more distant station in preference to a closer station.
SlotSel_Level2_A	To demonstrate that a station will select a slot at level 2 when the appropriate criteria are satisfied.
SlotSel_Level2_B	To demonstrate that a station will select a slot at level 2, excluding those slots not meeting the criteria of level 2 or any lower priority level.
SlotSel_Level2_C	To demonstrate that a station will select a slot at level 2 in preference to those available at level 3.
SlotSel_Level2_D	To demonstrate that a station will select a slot at level 2 in preference to those available at level 4.
SlotSel_Level2_E	To demonstrate that a station will select slots at level 2 from a more distant station in preference to a closer station.

Test Case Name	Description
SlotSel_Level3_A	To demonstrate that a station will select a slot at level 3 when the appropriate criteria are satisfied.
SlotSel_Level3_B	To demonstrate that a station will select a slot at level 3, excluding those slots not meeting the criteria of level 3 or any lower priority level.
SlotSel_Level3_C	To demonstrate that a station will select a slot at level 3 in preference to those available at level 4.
SlotSel_Level3_D	To demonstrate that a station will select slots at level 3 from a more distant station in preference to a closer station.
SlotSel_Level4_A	To demonstrate that a station will select a slot at level 4 when the appropriate criteria are satisfied.
SlotSel_Level4_B	To demonstrate that a station will select a slot at level 4, excluding those slots not meeting the criteria of level 4.
SlotSel_Level4_C	To demonstrate that a station will select a slot at level 4 from a more distant station in preference to a closer station.
SlotSel_Block_Level0_A	To demonstrate that a station will select a block of slots at level 0 when no slots are reserved.
SlotSel_Block_Level0_B	To demonstrate that a station will select a block of slots at level 0, excluding those not meeting the criteria of any other level.
SlotSel_Block_MixedLevel	To demonstrate that a station will select a block of slots from slots available at different levels.
SlotSel_Reselection	To demonstrate that a station after selecting a slot which has been reserved by another station will not select a slot which has been reserved by the same station within the next M1-1 slots.
SlotSel_Unsuccessful	To demonstrate that a station will fail to select a slot when no slots are available which are compatible with the QoS parameters.
SlotSel_QoSGroup	To demonstrate that a station will select a slot using a second group of QoS parameters when no slot has been selected by means of the first group.
SlotSel_Exclusion	To demonstrate that a station does not select a slot for transmission when the station is required to transmit in that slot on another channel.
Conflict_Periodic_A	To demonstrate that a station will continue to transmit a periodic stream without action in the event of a conflicting non-periodic transmission from another station.
Conflict_Periodic_B	To demonstrate that a station will dither a periodic stream to resolve a conflict with a periodic stream from another station.
Conflict_Periodic_C	To demonstrate that a station will move a periodic stream to a new location in the event of a conflict with a periodic stream from another station that does not allow the original stream to be dithered.
Conflict_NoAction	To demonstrate that a station will continue to transmit a periodic stream without action in the event of receiving a conflicting reservation such that the slot remains available.
Conflict_Incremental	To demonstrate that a station will not transmit in a slot previously reserved by an incremental broadcast reservation in the event of receiving a conflicting reservation, and will make the broadcast in an alternative slot by random access.
Conflict_BND	To demonstrate that a station which has made a BND reservation will not transmit in the reserved slot in the event of a conflicting reservation made by a periodic broadcast.
Conflict_Priority	To demonstrate that a station required to transmit in the same slot by conflicting requests will transmit the response of highest priority.
Conflict_FirstRequest	To demonstrate that a station required to transmit in the same slot by conflicting requests of equal priority will transmit the response to the first request.
Conflict_Channel_Priority	To demonstrate that a station required to transmit in the same slot on different channels by conflicting requests will transmit the response of highest priority.
Conflict_Channel_Ground_A	To demonstrate that when a station is requested to make a sync burst transmission in the same slot on different channels, the request from the ground station takes precedence.
Conflict_Channel_Ground_B	To demonstrate that when a station is requested to make a sync burst transmission in the same slot on different channels by more than one ground station, the latest request takes precedence.
Conflict_Channel_FirstRequest	To demonstrate that when a station is requested to make a sync burst transmission in the same slot on different channels by more than one ground station, the latest request takes precedence.
Slot_Boundary	To demonstrate that a transmission from the station complies with timing performance requirements at the slot boundary.
Slot_Delayed	To demonstrate that a delayed transmission from the station complies with timing performance requirements.

Test Case Name	Description
Rand_Busy	To demonstrate that a station will not make a random access transmission in a slot perceived to be busy at the start of the slot (e.g. a transmission which extends beyond the guard time).
Rand_Congestion	To demonstrate that the VSS User is informed if a request to make a random transmission is not successful within TM2 slots.
Rand_Persistence	To demonstrate that a random transmission is made with probability p.
Rand_MaxAttempts	To demonstrate that the station will authorize a random transmission as soon as the channel is available after VS3 unsuccessful attempts
Rand_Priority	To demonstrate that bursts queued for transmission by random access are transmitted in order of priority.
Rand_TM2Reset	To demonstrate that timer TM2 is reset following a successful random transmission when a further burst is queued for transmission.
Rand_TM2Clear	To demonstrate that timer TM2 is cleared following a successful random transmission when no further bursts are queued for transmission.
Rand_VS3Clear	To demonstrate that if a request to make a random transmission is not successful within TM2 slots then the VS3 counter is cleared and no transmission is made.
Rand_Availability	To demonstrate that a station makes random access attempts in slots available only at levels 0 to 2.
Null_Reservation	To demonstrate that no slot is reserved following the receipt of a null reservation.
Periodic_InitialRes	To demonstrate that in the absence of any conflicting reservation, a station will maintain a periodic reservation in a constant position in the superframe, with pt = 3 and po = 0, until announcing a further dither.
Periodic_NonDitherRes	To demonstrate that a station receiving a periodic broadcast reservation specifying no dither will reserve the appropriate slots.
Periodic_Potential	To demonstrate that a station's reservation table will also record potential reservations, defined as the M1, 2 M1, 3 M1 and 4 M1 slots after a slot, for which no transmission has been decoded.
Periodic_DitherRes	To demonstrate that a station receiving a periodic broadcast reservation specifying dither will reserve the appropriate slots.
Periodic_DitherRange	To demonstrate that a station will maintain a periodic stream within the dither range in accordance with the V11 and V12 parameters.
Periodic_DitherOffset_A	To demonstrate that in the absence of a conflicting reservation, a station will announce a dither to a periodic stream three superframes before the dither occurs.
Periodic_DitherOffset_B	To demonstrate that in the absence of a conflicting reservation, following announcement of a dither to a periodic stream, the same dithered slot will be reserved by each of the subsequent two transmissions, containing decrementing values of pt.
Periodic_DitherOffset_C	To demonstrate that a station will always dither away from the current transmission slot.
Periodic_DitherOffset_D	To demonstrate that following announcement of a dither to a periodic stream, the transmission slot will be adjusted to occupy the reserved slot.
Periodic_IndependentStreams	To demonstrate that separate streams of periodic broadcasts dither independently.
Periodic_Replacement	To demonstrate that a station receiving a periodic broadcast reservation in a slot previously by a periodic broadcast will replace the previous reservations by those carried in the new transmission.
Periodic_Quarantine_A	To demonstrate that a periodic reservation issued by a ground station will cause the following slots to be regarded as quarantined by a station receiving the response.
Periodic_Quarantine_B	To demonstrate that a periodic broadcast issued by a ground station greater than VS4 away will not cause slots to be quarantined.
Periodic_Availability_A	To demonstrate that a station will take account of the availability of the current transmission slot when dithering to a new slot.
Periodic_Availability_B	To demonstrate that when the current transmission slot is occupied at the dither of a periodic broadcast, the slot availability is determined from the first occupancy of the slot by a different station.
Periodic_Rate	To demonstrate that the station will establish a set of periodic streams at a nominal periodic rate according to the V11 parameter.
Periodic_TV11	To demonstrate that in the absence of any conflicting reservation a station will set the value of TV11 uniformly between the minimum and maximum values.
Periodic_Cancel	To demonstrate that a station receiving a periodic broadcast cancellation in a slot previously reserved for a periodic broadcast will cancel the periodic stream

Test Case Name	Description
Periodic_CancelIncremental	To demonstrate that upon receipt of an incremental broadcast in a slot expected to contain a periodic broadcast from the same peer station, the periodic stream is cancelled.
Periodic_CancelUnicast	To demonstrate that upon receipt of a unicast request with source/destination flag set to 1 in a slot expected to contain a periodic broadcast from the same peer station, the periodic stream is cancelled.
Periodic_CancelQuarantine	To demonstrate that a cancellation of a periodic reservation will cause quarantine to be preserved for one superframe following the cancellation.
Incremental_Reservation_A	To demonstrate that a station receiving an incremental broadcast reservation will reserve the appropriate slots.
Incremental_Reservation_B	To demonstrate that an incremental broadcast with io = 0 causes no reservation to be made.
Incremental_Request	To demonstrate that a station will select and reserve a series of future transmission slots by means of the incremental broadcast protocol.
Incremental_SlotSel	To demonstrate that a slot is selected for an incremental broadcast reservation from the appropriate candidate range.
Combined_Reservation	To demonstrate that receipt of a combined periodic and incremental broadcast reservation causes the appropriate slots to be reserved.
BND_Reservation	To demonstrate that reception of a BND reservation causes the appropriate slots to be reserved.
Unicast_Reservation_A	To demonstrate that reception of a point-to-point unicast reservation for the destination station to transmit causes the appropriate slots to be reserved.
Unicast_Reservation_B	To demonstrate that a reception of a point-to-point unicast reservation for the source station to transmit causes the appropriate slots to be reserved.
Unicast_Reservation_C	To demonstrate that a reception of a broadcast unicast reservation causes the appropriate slots to be reserved.
Unicast_Reservation_D	To demonstrate that a station applying the slot selection criteria will exclude any slot reserved by another station using the unicast request protocol with sdf = 1.
Info_Reservation	To demonstrate that a station receiving a burst containing an information transfer request reservation addressed to another station will reserve the slots identified for the information transfer and acknowledgement.
Autotune_Autonomous_A	To demonstrate that an autotune request for autonomous broadcasts on the current frequency with trmt = 0 causes a station to send autonomous sync bursts as directed, cancelling the current autonomous transmissions, and also that the default sync burst reporting rate is restored following expiry of the request.
Autotune_Autonomous_B	To demonstrate that an autotune request for autonomous broadcasts on a different frequency with trmt = 0 causes a station to send autonomous sync bursts as directed, cancelling the current autonomous transmissions, and also that the default sync burst reporting rate is restored following expiry of the request.
Autotune_Autonomous_C	To demonstrate that an autotune request for autonomous broadcasts with trmt = 1 causes a station to continue current autonomous transmissions during and after expiry of the request.
Autotune_Autonomous_D	To demonstrate that an autotune request for autonomous broadcasts on a different frequency with rcvr = 00 or 11 causes a station to continue receiving sync bursts on the current frequency.
Autotune_Autonomous_E	To demonstrate that an autotune request for autonomous broadcasts on a different frequency with rcvr = 01 causes a station to receive sync bursts on the indicated frequency.
Autotune_AutonomousCessation	To demonstrate that on cessation of an autotune command directing the reporting rate, normal default behaviour on the GSCs is established.
Autotune_Directed_A	To demonstrate that an autotune request directing transmission in specified slots on the current frequency, causes a station to transmit in the directed slots, cancelling current autonomous transitions, and that the default sync burst reporting rate is restored following expiry of the command.
Autotune_Directed_B	To demonstrate that an autotune request directing transmission in specified slots on a different frequency, causes a station to transmit in the directed slots, cancelling current autonomous transitions, and that the default sync burst reporting rate is restored following expiry of the command.
Autotune_Unsupported	To demonstrate that when an autotune request cannot be supported due to lack of receiver resources, a general failure is issued with the required error code.
Autotune_Reservation	To demonstrate that a station receiving a directed request from a ground station addressed to another station will reserve the directed slots.
Autotune_CancelResp	To demonstrate that a station making a directed request cancellation will provoke the addressed station to cancel the directed broadcast.

Test Case Name	Description
Autotune_CancelAbsent	To demonstrate that a station receiving a directed request addressed to another station will take no action upon receipt of a directed cancellation from the directing station alone.
Autotune_Override	To demonstrate that a station receiving a directed request with the override flag set will cancel previously placed directed request reservations made by the same station.
Autotune_Invalid_A	To demonstrate that an invalid directed rate autotune reservation results in no action.
Autotune_Invalid_B	To demonstrate that an invalid frequency field in an autotune reservation results in no reservation.
Autotune_DirectedCessation	To demonstrate that on cessation of an autotune command directing slots to use, normal default behaviour on the GSCs is established.
Autotune_DirectedCancelAutonomous	To demonstrate that receipt of a directed slot autotune causes autonomous reporting of the same data to be cancelled.
Autotune_DirectedQuarantine	To demonstrate that a response to a directed request issued by a ground station will cause the following slots to be regarded as quarantined by a station receiving the response.
PleaResponse_Reservation_A	To demonstrate that receipt of a plea response with a standard nominal rate causes the appropriate slots to be reserved.
PleaResponse_Reservation_B	To demonstrate that receipt of a plea response with a special nominal rate causes the appropriate slots to be reserved.
PleaResponse_Transmission_A	To demonstrate that receipt of a plea addressed to a station results in transmission of a plea response of the appropriate format.
PleaResponse_Transmission_B	To demonstrate that a second plea addressed to a station results in transmission of a plea response containing the remaining future slots from the previous plea response.
PleaResponse_Retransmission	To demonstrate that a plea response is not re-transmitted.
Block_Superframe_Reserve_A	To demonstrate that the station will not transmit in blocks of slots reserved by a ground station by means of a superframe block message.
Block_Superframe_Reserve_B	To demonstrate that the station will not transmit in blocks of slots reserved by a ground station by means of a superframe block message with block offset (bo) non-zero but block timeout (bt) set to 3.
Block_Superframe_Reserve_C	To demonstrate that the station will not transmit in blocks of slots reserved by a ground station by means of a superframe block message with block offset (bo) non-zero.
Block_Superframe_Invalid	To demonstrate that the station will ignore a superframe block reservation with re-broadcast offset < 2.
Block_Superframe_Rebroadcast	To demonstrate that a station will re-broadcast a superframe blocking message in accordance with a ground station request.
Block_Superframe_Quarantine	To demonstrate that a block reservation message issued by a station greater than distance VS4 away will not cause slots to be reserved.
Block_Secondframe_Reserve	To demonstrate that the station will not transmit in blocks of slots reserved by a ground station by means of a second frame block message.
Response_Reservation	To demonstrate that a response reservation field is recognized and causes no reservation to be made.
Request_Unsupported	To demonstrate that a station will respond to a general request burst that cannot be supported with a general failure burst.
Sync_Format	To demonstrate that a station will broadcast a sync burst with the correct format.
Sync_Format_Rec	To demonstrate that a station will correctly process a received sync burst.
Sync_Latency	To demonstrate that the field data age of the synchronization burst is correctly set.
Sync_Rate	To demonstrate that the station outputs autonomous sync bursts at a rate of at least one per M1 slots on all Global Signalling Channels (GSCs).
Sync_Interval	To demonstrate that a station outputs autonomous sync bursts with a uniform interval between nominal slots on each GSC.
Sync_Fixed_NIC	To demonstrate that a station sets the navigation uncertainty category appropriately.
Sync_Fixed_BaseAlt	To demonstrate that a station sets the base altitude in the fixed part of the sync burst in accordance with the input altitude data.
Sync_Fixed_DataAge	To demonstrate that a station sets the data age subfield of a sync burst appropriately.
NetEntry_Periodic	To demonstrate that a station which desires to gain entry to a network using the combined periodic and incremental broadcast protocols is able to set up a series of regularly spaced streams.

Test Case Name	Description
NetEntry_Delayed_Plea	To demonstrate that a station which desires to perform network entry using a delayed plea transmission will make such a transmission in an otherwise unoccupied slot.
NetEntry_Delayed_BND	To demonstrate that a station which desires to perform network entry using a delayed BND transmission will make such a transmission in an otherwise unoccupied slot.
NetEntry_Receive	To demonstrate that a station in receipt of a delayed transmission containing a plea will generate a reply to the source station with slots for it to transmit in, if it has some slots which it could make available.
NetEntry_OneMinute	To demonstrate that a station which desires to transmit for the first time without using network entry protocols, will listen to the channel on which it desires to transmit for M1 + 126 slots prior to making any transmissions.
NetEntry_Repeat	To demonstrate that a station does not repeat net entry within M1 slots following the previous successful net entry.
UCTRL_param_VS1	To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter VS1.
UCTRL_param_VS2	To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter VS2.
UCTRL_param_VS4	To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter VS4.
UCTRL_param_Q4	To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter Q4.
UCTRL_param_CG1_limit	To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter CG1_limit.
UCTRL_param_CG1_range	To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter CG1_range.
UCTRL_param_CG1_inc	To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter CG1_inc.
Param_CG1_reach	To demonstrate that a station will act correctly upon modification of parameter CG1_inc.
UCTRL_param_CG1_decay	To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter CG1_decay.
UCTRL_param_M2inc	To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter M2inc.
UCTRL_param_M2limit	To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter M2limit.
Sync_Report_Receive	To demonstrate that a station receiving a sequence of sync burst reports from a peer station will generate an appropriate output.
Sync_Report_Simultaneous	To demonstrate that a station is capable of receiving sync burst reports simultaneously on both GSCs.
CPR_Encode	To demonstrate that a series of latitude and longitude positions may be correctly encoded in the sync burst using the CPR algorithm.
CPR_Decode	To demonstrate that a series of latitude and longitude positions may be correctly decoded from the sync burst using the CPR algorithm.
Power_Interrupt_A	To demonstrate that a station continues to operate the receiver through a short power interrupt not exceeding Y ms.
Power_Interrupt_B	To demonstrate that a station will observe the m2 filter after a power interrupt exceeding Y ms.
DLS_NotSupported	To demonstrate that a station in receipt of a CTRL_RTS transmits a general failure with an error type of 80 hex when it does not support the DLS.
DLS_UDATA_Receive	To demonstrate that a UDATA DLPDU received from another station will be forwarded to the DLS user.
DLS_UDATA_Invalid	To demonstrate that a station receiving an invalid UDATA DLPDU will detect and discard it.
Param_L1	To demonstrate that a station displays correct operation of counter L1.

If a station will use network entry by plea/response procedures then it shall conform to the test NetEntry_Delayed_Plea. Otherwise, it shall not be required to conform to this test.

If a station will use network entry by BND then it shall conform to the test NetEntry_Delayed_BND. Otherwise, it shall not be required to conform to this test.

If a station will use network entry by listening to the channel for M1 + 126 slots followed by full-slot random transmission with combined periodic/incremental reservation types, then it shall conform to the test NetEntry_Periodic. Otherwise, it shall not be required to conform to this test.

7.4.2 Declarations

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

7.4.3 Constraints

7.4.3.1 Abbreviations

7.4.3.1.1 Subfield mnemonics

Table 7.2: Subfield mnemonics

Mnemonic	Meaning
a	Additional slots
a/d	Autonomous/directed flag
auto	Autonomous information
balt	Base altitude
b/g	Baro/geo altitude
blg	Block length
bo	Block offset
br	Block repeat rate
bs	Block start
bt	Block timeout
c	CRC
cprf	CPR format even/odd
d	Destination address
da	Data age
dos	Directory of services flag
erid	Extended reservation ID
f	Frequency
flag	Flag delimiting burst
id	Information field identity
in	Information field
io	Incremental offset
lat	Latitude
lon	Longitude
mi	Message ID
nd	Negative dither
nr	Nominal update rate
nic	Position navigation integrity category
off	Offset to first reserved slot
ok	Confirm/failure flag
po	Periodic offset
pr	Priority
pr_flag	Plea response flag
pt	Periodic timeout
r-b/a	Requested baro/geo altitude
r-mi	Requested message ID
rcvr	Receiver control
rd	Reservation data
res	Reserved bit
rid	Reservation ID
ro	Response offset
roff	Re-broadcast offset
s	Source address
sdf	Source/destination flag

Mnemonic	Meaning
sleep	Autonomous monitoring
sz	Size
tc	Trajectory Change Point change flag
tfom	Time FOM
ver	Version number
vt	Timeout

7.4.3.1.2 Special characters used in the subfield definitions

Table 7.3: Special characters used in the subfield definitions

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

7.4.3.1.3 Station addresses and positions

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

add_A = address of 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, defined in clause 5.6 of ETSI Ground Station.

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 positions of the simulated stations in the tests have been given on the basis of the following set of values of the Q2 parameters. This set is used as the default in the tests and referred to as Q2 Set 1.
The ICAO VDL4 Technical Manual [1] default values are used for the parameters Q2a, Q2b, and Q2d. A value of Q2c = 120 NM is used in order to allow testing of conditions it would not be possible to test if Q2c = 0.

Table 7.4: Q2 Parameters: Q2 Set 1 (Default for all tests)

Symbol	Parameter Name	Value	Notes
Q2a	Slot selection range constraint for level 1	150 NM	
Q2b	Slot selection range constraint for level 2	150 NM	
Q2c	Slot selection range constraint for level 3	120 NM	Not a VDL4 Technical Manual default value
Q2d	Slot selection range constraint for level 4	300 NM	

For some tests, a second less stringent set of values for the Q2 parameters is specified, to be used when slot selection fails using Set 1. This set is defined below and referred to as Q2 Set 2.

Table 7.5: Q2 Parameters: Q2 Set 2

Symbol	Parameter Name	Value
Q2a	Slot selection range constraint for level 1	100 NM
Q2b	Slot selection range constraint for level 2	100 NM
Q2c	Slot selection range constraint for level 3	80 NM
Q2d	Slot selection range constraint for level 4	200 NM

For some tests, the following set of values for the Q2 parameters is used.

Table 7.6: Q2 Parameters: Q2 Set 3

Symbol	Parameter Name	Value
Q2a	Slot selection range constraint for level 1	150 NM
Q2b	Slot selection range constraint for level 2	150 NM
Q2c	Slot selection range constraint for level 3	120 NM
Q2d	Slot selection range constraint for level 4	120 NM

For assessment of conflict resolution, the following set of values for the Q2 parameters is used, as specified in ICAO VDL4 Technical Manual [1], clause 1.5.5.1.4.

Table 7.7: Q2 Parameters: Q2 Set 4

Symbol	Parameter Name	Value
Q2a	Slot selection range constraint for level 1	360 NM
Q2b	Slot selection range constraint for level 2	360 NM
Q2c	Slot selection range constraint for level 3	360 NM
Q2d	Slot selection range constraint for level 4	360 NM

7.4.3.1.4 Tables of values for use in CPR test cases

The following tables are used in the CPR test cases.

Table 7.8: CPR state machine for position report processing (points not straddling a transition level)

In State				**1**	**2**		**3**		**4**	
Last report				**None**	**Even**	**Odd**	**Even**	**Odd**	**Even**	**Odd**
Target position quality				**None**	**None**	**None**	**Local**	**Local**	**Global**	**Global**
Received position report type	**Own Posi-tion**	**Timers (exp = expired)**								
		TR1	**TR2**							
Even or odd with patch ID	not appli-cable	not appli-cable	not appli-cable	Op= 1a N = 4 C = GL resTR1 resTR2	Op= 2a N = 4, C = GL resTR1, resTR2		Op= 3a N = 4, C = GL resTR1, resTR2		Op= 4a N = 4, C = GL resTR1, resTR2	
Even	Yes	Not exp	Not exp	Op= 1b N = 3 C = L1 resTR1	Op= 2b N = 3 C = L1 resTR1	Op= 2c N = 4 C = GL resTR1 resTR2	Op= 3b N = 3 C = L1 resTR1	Op= 3c N = 4 C = GL resTR1 resTR2	Op= 4b N = 4 C = L2 resTR1	Op= 4d N = 4 C = GL resTR1 resTR2
			Exp						Op= 4c N = 3 C = L1 resTR1	
		Exp	Not exp			Op= 2d N = 3 C = L1 resTR1		Op= 3d N = 3 C = L1 resTR1	Op= 4e N = 4, C = L2, resTR1	
			Exp						Op= 4f N = 3, C = L1, resTR1	
	No	Not exp	Not exp	Op= 1c N = 2 C = NO resTR1	Op= 2e N = 2 C = NO resTR1	Op= 2f N = 4 C = GL resTR1 resTR2	Op= 3e N = 2 C = NO resTR1	Op= 3f N = 4 C = GL resTR1 resTR2	Op= 4g N = 4 C = L2 resTR1	Op= 4i N = 4 C = GL resTR1 resTR2
			Exp						Op= 4h N = 2 C = NO resTR1	
		Exp	Not exp			Op= 2g N = 2 C = NO resTR1		Op= 3g N = 2 C = NO resTR1	Op= 4j N = 4, C = L2, resTR1	
			Exp						Op= 4k N = 2, C = NO, resTR1	

<table>
<tr><th colspan="4">In State</th><th>1</th><th colspan="2">2</th><th colspan="2">3</th><th colspan="2">4</th></tr>
<tr><th colspan="4">Last report</th><th>None</th><th>Even</th><th>Odd</th><th>Even</th><th>Odd</th><th>Even</th><th>Odd</th></tr>
<tr><th colspan="4">Target position quality</th><th>None</th><th>None</th><th>None</th><th>Local</th><th>Local</th><th>Global</th><th>Global</th></tr>
<tr><th rowspan="2">Received position report type</th><th rowspan="2">Own Posi-tion</th><th colspan="2">Timers (exp = expired)</th><th rowspan="2"></th><th rowspan="2"></th><th rowspan="2"></th><th rowspan="2"></th><th rowspan="2"></th><th rowspan="2"></th><th rowspan="2"></th></tr>
<tr><th>TR1</th><th>TR2</th></tr>
<tr><td rowspan="8">Odd</td><td rowspan="4">Yes</td><td rowspan="2">Not exp</td><td>Not exp</td><td rowspan="4">Op= 1d
N = 3
C = L1
resTR1</td><td rowspan="2">Op= 2h
N = 4
C = GL
resTR1
resTR2</td><td rowspan="4">Op= 2j
N = 3
C = L1
resTR1</td><td rowspan="2">Op= 3h
N = 4
C = GL
resTR1
resTR2</td><td rowspan="4">Op= 3j
N = 3
C = L1
resTR1</td><td rowspan="2">Op= 4l
N = 4
C = GL
resTR1
resTR2</td><td>Op= 4m
N = 4
C = L2
resTR1</td></tr>
<tr><td>Exp</td><td>Op= 4n
N = 3
C = L1
resTR1</td></tr>
<tr><td rowspan="2">Exp</td><td>Not exp</td><td rowspan="2">Op= 2i
N = 3
C = L1
resTR1</td><td rowspan="2">Op= 3i
N = 3
C = L1
resTR1</td><td colspan="2">Op= 4o
N = 4, C = L2,
resTR1</td></tr>
<tr><td>Exp</td><td colspan="2">Op= 4p
N = 3, C = L1,
resTR1</td></tr>
<tr><td rowspan="4">No</td><td rowspan="2">Not exp</td><td>Not exp</td><td rowspan="4">Op= 1e
N = 2
C = NO
resTR1</td><td rowspan="2">Op= 2k
N = 4
C = GL
resTR1
resTR2</td><td rowspan="4">Op= 2m
N = 2
C = NO
resTR1</td><td rowspan="2">Op= 3k
N = 4
C = GL
resTR1
resTR2</td><td rowspan="4">Op= 3m
N = 2
C = NO
resTR1</td><td rowspan="2">Op= 4q
N = 4
C = GL
resTR1
resTR2</td><td>Op= 4r
N = 4
C = L2
resTR1</td></tr>
<tr><td>Exp</td><td>Op= 4s
N = 2
C = NO
resTR1</td></tr>
<tr><td rowspan="2">Exp</td><td>Not exp</td><td rowspan="2">Op= 2l
N = 2
C = NO
resTR1</td><td rowspan="2">Op= 3l
N = 2
C = NO
resTR1</td><td colspan="2">Op= 4t
N = 4, C = L2,
resTR1</td></tr>
<tr><td>Exp</td><td colspan="2">Op= 4u
N = 2, C = NO,
resTR1</td></tr>
</table>

Table 7.9: CPR state machine for position report processing (transition level straddling)

<table>
<tr><th colspan="4">In State</th><th>1</th><th colspan="2">2</th><th colspan="2">3</th><th colspan="2">4</th></tr>
<tr><th colspan="4">Last report</th><th>None</th><th>Even</th><th>Odd</th><th>Even</th><th>Odd</th><th>Even</th><th>Odd</th></tr>
<tr><th colspan="4">Target position quality</th><th>None</th><th>None</th><th>None</th><th>Local</th><th>Local</th><th>Global</th><th>Global</th></tr>
<tr><th rowspan="2">Received position report type</th><th rowspan="2">Own Posi-tion</th><th colspan="2">Timers (exp = expired)</th><th rowspan="2"></th><th rowspan="2"></th><th rowspan="2"></th><th rowspan="2"></th><th rowspan="2"></th><th rowspan="2"></th><th rowspan="2"></th></tr>
<tr><th>TR1</th><th>TR2</th></tr>
<tr><td rowspan="4">Even</td><td rowspan="2">Yes</td><td rowspan="2">Not exp</td><td>Not exp</td><td rowspan="4">See previous table</td><td rowspan="4">See previous table</td><td rowspan="2">Op= 2n
N = 3
C = L1
resTR1</td><td rowspan="4">See table 7.8</td><td rowspan="2">Op= 3n
N = 3
C = L1
resTR1</td><td rowspan="4">See table 7.8</td><td>Op= 4v
N = 4
C = L2
resTR1</td></tr>
<tr><td>Exp</td><td>Op= 4w
N =3
C = L1
res TR1</td></tr>
<tr><td rowspan="2">No</td><td rowspan="2">Not exp</td><td>Not exp</td><td rowspan="2">Op= 2o
N = 2
C = NO
resTR1</td><td rowspan="2">Op= 3o
N = 2
C = NO
resTR1</td><td>Op= 4x
N = 4
C = L2
resTR1</td></tr>
<tr><td>Exp</td><td>Op= 4y
N = 2
C = NO
resTR1</td></tr>
<tr><td rowspan="4">Odd</td><td rowspan="2">Yes</td><td rowspan="2">Not exp</td><td>Not exp</td><td rowspan="4">See previous table</td><td rowspan="2">Op= 2p
N = 3
C = L1
resTR1</td><td rowspan="4">See previous table</td><td rowspan="2">Op= 3p
N = 3
C = L1
resTR1</td><td rowspan="4">See previous table</td><td>Op= 4z
N = 4
C = L2
resTR1</td><td rowspan="4">See previous table</td></tr>
<tr><td>Exp</td><td>Op= 4za
N =3
C =L1
resTR1</td></tr>
<tr><td rowspan="2">No</td><td rowspan="2">Not exp</td><td>Not exp</td><td rowspan="2">Op= 2q
N = 2
C = NO
resTR1</td><td rowspan="2">Op= 3q
N = 2
C = NO
resTR1</td><td>Op= 4zb
N = 4
C = L2
resTR1</td></tr>
<tr><td>Exp</td><td>Op= 4zc
N = 2
C = NO
resTR1</td></tr>
</table>

Table 7.10: 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

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

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

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 s. 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 below (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 7.11: Test values for CPR report encoding CPR_ENC_TABLE (row, column) (CE(r, c))

latitude	longitude	cpr_type	lat_enc	lon_enc
12,855700000	-0,815000000	0	1 169	15 085
12,872000000	-0,798700000	1	1 030	15 147
12,888300000	-0,782400000	0	1 183	15 137
12,904600000	-0,766100000	1	1 043	15 198
12,920900000	-0,749800000	0	1 196	15 189
12,953500000	-0,717200000	0	1 209	15 241
12,969800000	-0,700900000	1	1 069	15 299
13,002400000	-0,668300000	1	1 082	15 349
13,051300000	-0,619400000	0	1 250	15 396
13,100200000	-0,570500000	1	1 121	15 500
13,165400000	-0,505300000	1	1 146	15 601
13,279500000	-0,391200000	0	1 343	15 760
13,312100000	-0,358600000	0	1 356	15 812
13,328400000	-0,342300000	1	1 211	15 853
13,409900000	-0,260800000	0	1 396	15 968
13,426200000	-0,244500000	1	1 250	16 005
13,442500000	-0,228200000	0	1 410	16 020
13,475100000	-0,195600000	0	1 423	16 071
13,491400000	-0,179300000	1	1 276	16 106
13,507700000	-0,163000000	0	1 436	16 123
13,524000000	-0,146700000	1	1 289	16 163
13,540300000	-0,130400000	0	1 450	16 181
13,556600000	-0,114100000	1	1 302	16 212
13,589200000	-0,081500000	1	1 315	16 261
13,654400000	-0,016300000	1	1 341	16 359
13,735900000	0,065200000	0	1 530	101
13,850000000	0,179300000	1	1 419	269
13,866300000	0,195600000	0	1 583	303
13,996700000	0,326000000	0	1 637	504
14,013000000	0,342300000	1	1 484	514
14,143400000	0,472700000	1	1 536	710
14,208600000	0,537900000	1	1 562	808
14,290100000	0,619400000	0	1 757	958
14,322700000	0,652000000	0	1 770	1 009
14,404200000	0,733500000	1	1 640	1 102
14,436800000	0,766100000	1	1 653	1 151
14,518300000	0,847600000	0	1 850	1 311
14,550900000	0,880200000	0	1 864	1 362
14,632400000	0,961700000	1	1 731	1 444

latitude	longitude	cpr_type	lat_enc	lon_enc
14,713900000	1,043200000	0	1 930	1 614
14,746500000	1,075800000	0	1 944	1 665
14,762800000	1,092100000	1	1 782	1 640
14,876900000	1,206200000	0	1 997	1 866
14,893200000	1,222500000	1	1 834	1 836
15,007300000	1,336600000	0	2 050	2 068
15,088800000	1,418100000	1	1 912	2 130
15,121400000	1,450700000	1	1 925	2 179
15,154000000	1,483300000	1	1 938	2 228
15,235500000	1,564800000	0	2 144	2 421
15,268100000	1,597400000	0	2 157	2 472
15,349600000	1,678900000	1	2 016	2 521
15,365900000	1,695200000	0	2 197	2 623
15,480000000	1,809300000	1	2 068	2 717
15,496300000	1,825600000	0	2 251	2 825
15,610400000	1,939700000	1	2 120	2 913
15,626700000	1,956000000	0	2 304	3 026
15,740800000	2,070100000	1	2 172	3 109
15,757100000	2,086400000	0	2 358	3 228
15,871200000	2,200500000	1	2 224	3 305
15,887500000	2,216800000	0	2 411	3 430
15,903800000	2,233100000	1	2 237	3 354
15,920100000	2,249400000	0	2 424	3 480
19,098600000	5,427900000	1	3 509	8 151
19,114900000	5,444200000	0	3 733	8 424
19,131200000	5,460500000	1	3 522	8 200
19,147500000	5,476800000	0	3 746	8 474
19,163800000	5,493100000	1	3 535	7 999
19,180100000	5,509400000	0	3 759	8 274
23,483300000	9,812600000	1	1 159	14 290
23,499600000	9,828900000	0	1 433	14 761
23,515900000	9,845200000	1	1 172	14 337
23,532200000	9,861500000	0	1 446	14 361
23,548500000	9,877800000	1	1 185	13 935
27,167100000	13,496400000	0	2 935	3 271
27,183400000	13,512700000	1	2 632	2 680
27,199700000	13,529000000	0	2 948	3 319
27,216000000	13,545300000	1	2 645	2 726
27,232300000	13,561600000	0	2 962	2 749
27,248600000	13,577900000	1	2 658	2 154
30,361900000	16,691200000	0	148	7 164
30,378200000	16,707500000	1	3 904	6 427
30,394500000	16,723800000	0	162	7 210
30,508600000	16,837900000	1	3 956	6 605
30,524900000	16,854200000	0	215	6 627
30,541200000	16,870500000	1	3 969	5 882
30,557500000	16,886800000	0	228	6 672
33,361100000	19,690400000	0	1 376	10 499
33,377400000	19,706700000	1	1 003	9 625
33,393700000	19,723000000	0	1 390	10 544
33,507800000	19,837100000	1	1 055	9 797
33,524100000	19,853400000	0	1 443	9 818
33,540400000	19,869700000	1	1 068	8 936
33,556700000	19,886000000	0	1 456	9 861
36,132100000	22,461400000	1	2 100	12 238
36,148400000	22,477700000	0	2 518	13 282
36,164700000	22,494000000	1	2 113	12 280
36,278800000	22,608100000	0	2 571	13 454

latitude	**longitude**	**cpr_type**	**lat_enc**	**lon_enc**
36,295100000	22,624400000	1	2 165	11 416
36,311400000	22,640700000	0	2 585	12 467
36,327700000	22,657000000	1	2 178	11 456
38,723800000	25,053100000	1	3 132	14 400
38,740100000	25,069400000	0	3 579	15 561
38,756400000	25,085700000	1	3 145	14 440
38,870500000	25,199800000	0	3 632	15 727
38,886800000	25,216100000	1	3 197	13453
38,903100000	25,232400000	0	3 646	14 621
38,919400000	25,248700000	1	3 210	13 492
41,185100000	27,514400000	0	485	1 042
41,201400000	27,530700000	1	23	16 192
41,217700000	27,547000000	0	499	1 082
41,331800000	27,661100000	1	75	16 346
41,348100000	27,677400000	0	552	16 365
41,364400000	27,693700000	1	88	15 124
41,380700000	27,710000000	0	565	21
43,532300000	29,861600000	0	1 446	2 567
43,548600000	29,877900000	1	958	1 226
43,564900000	29,894200000	0	1 460	2 605
43,679000000	30,008300000	1	1 010	1 375
43,695300000	30,024600000	0	1 513	1 393
43,711600000	30,040900000	1	1 023	45
43,727900000	30,057200000	0	1 527	1 430
45,781700000	32,111000000	1	1 847	2 306
45,798000000	32,127300000	0	2 374	3 785
45,814300000	32,143600000	1	1 860	2 341
45,928400000	32,257700000	0	2 428	3 934
45,944700000	32,274000000	1	1 912	1 015
45,961000000	32,290300000	0	2 441	2 501
45,977300000	32,306600000	1	1 925	1 049
47,949600000	34,278900000	1	2 710	3 113
47,965900000	34,295200000	0	3 262	4 691
47,982200000	34,311500000	1	2 723	3 148
48,096300000	34,425600000	0	3 315	4 834
48,112600000	34,441900000	1	2 775	1 717
48,128900000	34,458200000	0	3 329	3 301
48,145200000	34,474500000	1	2 788	1 749

CPR test value tolerances

The number of decimal places afforded to the decoded latitude and longitude values in the last eight columns of the previous tables varies according to the resolution expected from the decoding algorithm (see table 7.12). When using the tables 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 the table 7.12.

Table 7.12: 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)	5	±0,0003
NOTE: These figures 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: Since the figures given in the previous tables 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.

Key to CPR decoding table in following clause.

Table 7.13: 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
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 state machine tables 7.8 and 7.9)
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

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

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

The encoding for the latitude and longitude values contained in the first four columns of the decoding table below is given in the encoding table above.

The decoding operation used in the table below is determined by the state machine tables, 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 must 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 7.14: 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		tim	o	op	cal	f	decoded lat	decoded lon
12,9	-0,8	12,855700000	-	-0,815000000	-	0	-	-	1		-	n	1c	NO	2	NO CALC	NO CALC
12,9	-0,8	12,872000000	12,855700000	-0,798700000	-0,815000000	1	0	0	2		10	n	2k	GL	4	12,8728	-0,79882
12,9	-0,8	12,888300000	12,872000000	-0,782400000	-0,798700000	0	1	0	4		10	n	4i	GL	4	12,8889	-0,78227
12,9	-0,8	12,904600000	12,888300000	-0,766100000	-0,782400000	1	0	0	4		10	n	4q	GL	4	12,9055	-0,76586
12,9	-0,7	12,920900000	12,904600000	-0,749800000	-0,766100000	0	1	0	4		10	n	4i	GL	4	12,9206	-0,74963
13,0	-0,7	12,953500000	12,920900000	-0,717200000	-0,749800000	0	0	0	4		20	n	4g	L2	4	12,9524	-0,71698
13,0	-0,7	12,969800000	12,953500000	-0,700900000	-0,717200000	1	0	0	4		10	n	4q	GL	4	12,9708	-0,70058
13,0	-0,7	13,002400000	12,969800000	-0,668300000	-0,700900000	1	1	0	4		20	n	4r	L2	4	13,0035	-0,66827
13,1	-0,6	13,051300000	13,002400000	-0,619400000	-0,668300000	0	1	0	4		30	n	4i	GL	4	13,0525	-0,61967
13,1	-0,6	13,100200000	13,051300000	-0,570500000	-0,619400000	1	0	0	4		30	n	4q	GL	4	13,1014	-0,57068
13,2	-0,5	13,165400000	13,100200000	-0,505300000	-0,570500000	1	1	0	4		40	n	4t	L2	4	13,1642	-0,50540
13,3	-0,4	13,279500000	13,165400000	-0,391200000	-0,505300000	0	1	0	4		70	n	4k	NO	2	NO CALC	NO CALC
13,3	-0,4	13,312100000	13,279500000	-0,358600000	-0,391200000	0	0	0	2		20	n	2e	NO	2	NO CALC	NO CALC
13,3	-0,3	13,328400000	13,312100000	-0,342300000	-0,358600000	1	0	0	2		10	n	2k	GL	4	13,3275	-0,34254
13,4	-0,3	13,409900000	13,328400000	-0,260800000	-0,342300000	0	1	0	4		50	n	4j	L2	4	13,4090	-0,26055
13,4	-0,2	13,426200000	13,409900000	-0,244500000	-0,260800000	1	0	0	4		10	y	4l	GL	4	13,4254	-0,24430
13,4	-0,2	13,442500000	13,426200000	-0,228200000	-0,244500000	0	1	0	4		10	y	4d	GL	4	13,4432	-0,22790
13,5	-0,2	13,475100000	13,442500000	-0,195600000	-0,228200000	0	0	0	4		10	y	4b	L2	4	13,4750	-0,19588
13,5	-0,2	13,491400000	13,475100000	-0,179300000	-0,195600000	1	0	0	4		10	y	4l	GL	4	13,4907	-0,17902
13,5	-0,2	13,507700000	13,491400000	-0,163000000	-0,179300000	0	1	0	4		10	y	4d	GL	4	13,5067	-0,16324
13,5	-0,1	13,524000000	13,507700000	-0,146700000	-0,163000000	1	0	1	4		10	y	4z	L2	4	13,5234	-0,14649
13,5	-0,1	13,540300000	13,524000000	-0,130400000	-0,146700000	0	1	0	4		10	y	4d	GL	4	13,5409	-0,13055
13,6	-0,1	13,556600000	13,540300000	-0,114100000	-0,130400000	1	0	0	4		10	y	4l	GL	4	13,5560	-0,11387
13,6	-0,1	13,589200000	13,556600000	-0,081500000	-0,114100000	1	1	0	4		10	y	4m	L2	4	13,5887	-0,08124
13,7	0,0	13,654400000	13,589200000	-0,016300000	-0,081500000	1	1	0	4		40	y	4o	L2	4	13,6540	-0,01598
13,7	0,1	13,735900000	13,654400000	0,065200000	-0,016300000	0	1	0	4		50	y	4e	L2	4	13,7363	0,06528
13,9	0,2	13,850000000	13,735900000	0,179300000	0,065200000	1	0	0	4		70	y	4p	L1	3	13,8499	0,17912
13,9	0,2	13,866300000	13,850000000	0,195600000	0,179300000	0	1	0	3		10	y	3c	GL	4	13,8657	0,19583
14,0	0,3	13,996700000	13,866300000	0,326000000	0,195600000	0	0	0	4		80	y	4f	L1	3	13,9976	0,32573
14,0	0,3	14,013000000	13,996700000	0,342300000	0,326000000	1	0	0	3		10	y	3h	GL	4	14,0132	0,34226

lat sut	lon sut	lat last	lat prev	lon last	lon prev	tl	tp	tr	i		tim	o	op	cal	f	decoded lat	decoded lon
14,1	0,5	14,143400000	14,013000000	0,472700000	0,342300000	1	1	0	4		80	n	4u	NO	2	NO CALC	NO CALC
14,2	0,5	14,208600000	14,143400000	0,537900000	0,472700000	1	1	0	2		40	n	2m	NO	2	NO CALC	NO CALC
14,3	0,6	14,290100000	14,208600000	0,619400000	0,537900000	0	1	0	2		50	y	2d	L1	3	14,2906	0,61915
14,3	0,7	14,322700000	14,290100000	0,652000000	0,619400000	0	0	0	3		20	y	3b	L1	3	14,3223	0,65211
14,4	0,7	14,404200000	14,322700000	0,733500000	0,652000000	1	0	0	3		50	y	3i	L1	3	14,4050	0,73380
14,4	0,8	14,436800000	14,404200000	0,766100000	0,733500000	1	1	0	3		20	y	3j	L1	3	14,4377	0,76643
14,5	0,8	14,518300000	14,436800000	0,847600000	0,766100000	0	1	0	3		50	y	3d	L1	3	14,5177	0,84729
14,6	0,9	14,550900000	14,518300000	0,880200000	0,847600000	0	0	0	3		20	n	3e	NO	2	NO CALC	NO CALC
14,6	1,0	14,632400000	14,550900000	0,961700000	0,880200000	1	0	0	2		50	n	2l	NO	2	NO CALC	NO CALC
14,7	1,0	14,713900000	14,632400000	1,043200000	0,961700000	0	1	0	2		50	n	2g	NO	2	NO CALC	NO CALC
14,7	1,1	14,746500000	14,713900000	1,075800000	1,043200000	0	0	0	2		20	y	2b	L1	3	14,7473	1,07608
14,8	1,1	14,762800000	14,746500000	1,092100000	1,075800000	1	0	0	3		10	n	3k	GL	4	14,7617	1,09204
14,9	1,2	14,876900000	14,762800000	1,206200000	1,092100000	0	1	0	4		70	n	4k	NO	2	NO CALC	NO CALC
14,9	1,2	14,893200000	14,876900000	1,222500000	1,206200000	1	0	0	2		10	y	2h	GL	4	14,8923	1,22255
15,0	1,3	15,007300000	14,893200000	1,336600000	1,222500000	0	1	0	4		70	n	4k	NO	2	NO CALC	NO CALC
15,1	1,4	15,088800000	15,007300000	1,418100000	1,336600000	1	0	0	2		50	y	2i	L1	3	15,0882	1,41832
15,1	1,5	15,121400000	15,088800000	1,450700000	1,418100000	1	1	0	3		20	n	3m	NO	2	NO CALC	NO CALC
15,2	1,5	15,154000000	15,121400000	1,483300000	1,450700000	1	1	0	2		20	y	2j	L1	3	15,1535	1,48358
15,2	1,6	15,235500000	15,154000000	1,564800000	1,483300000	0	1	0	3		50	n	3g	NO	2	NO CALC	NO CALC
15,3	1,6	15,268100000	15,235500000	1,597400000	1,564800000	0	0	0	2		20	y	2b	L1	3	15,2674	1,59764
15,3	1,7	15,349600000	15,268100000	1,678900000	1,597400000	1	0	0	3		50	n	3l	NO	2	NO CALC	NO CALC
15,4	1,7	15,365900000	15,349600000	1,695200000	1,678900000	0	1	0	2		10	n	2f	GL	4	15,3651	1,69523
15,5	1,8	15,480000000	15,365900000	1,809300000	1,695200000	1	0	0	4		70	n	4u	NO	2	NO CALC	NO CALC
15,5	1,8	15,496300000	15,480000000	1,825600000	1,809300000	0	1	0	2		10	y	2c	GL	4	15,4969	1,82578
15,6	1,9	15,610400000	15,496300000	1,939700000	1,825600000	1	0	0	4		70	y	4p	L1	3	15,6107	1,93970
15,6	2,0	15,626700000	15,610400000	1,956000000	1,939700000	0	1	0	3		10	n	3f	GL	4	15,6264	1,95569
15,7	2,1	15,740800000	15,626700000	2,070100000	1,956000000	1	0	0	4		70	n	4u	NO	2	NO CALC	NO CALC
15,8	2,1	15,757100000	15,740800000	2,086400000	2,070100000	0	1	0	2		10	n	2f	GL	4	15,7582	2,08624
15,9	2,2	15,871200000	15,757100000	2,200500000	2,086400000	1	0	0	4		70	y	4p	L1	3	15,8719	2,20073
15,9	2,2	15,887500000	15,871200000	2,216800000	2,200500000	0	1	0	3		10	n	3f	GL	4	15,8877	2,21679
15,9	2,2	15,903800000	15,887500000	2,233100000	2,216800000	1	0	0	4		10	n	4g	GL	4	15,9046	2,23336
15,9	2,2	15,920100000	15,903800000	2,249400000	2,233100000	0	1	0	4		10	n	4i	GL	4	15,9194	2,24910
19,1	5,4	19,098600000	-	5,427900000	-	1	-	-	1		-	n	1e	NO	2	NO CALC	NO CALC
19,1	5,4	19,114900000	19,098600000	5,444200000	5,427900000	0	1	0	2		10	n	2f	GL	4	19,1160	5,44438
19,1	5,5	19,131200000	19,114900000	5,460500000	5,444200000	1	0	0	4		10	n	4q	GL	4	19,1322	5,46021
19,1	5,5	19,147500000	19,131200000	5,476800000	5,460500000	0	1	0	4		10	n	4i	GL	4	19,1477	5,47670
19,2	5,5	19,163800000	19,147500000	5,493100000	5,476800000	1	0	1	4		10	n	4zb	L2	4	19,1648	5,49281
19,2	5,5	19,180100000	19,163800000	5,509400000	5,493100000	0	1	0	4		10	n	4i	GL	4	19,1795	5,50948

lat sut	lon sut	lat last	lat prev	lon last	lon prev	tl	tp	tr	i		tim	o	op	cal	f	decoded lat	decoded lon
23,5	9,8	23,483300000	-	9,812600000	-	1	-	-	1		-	y	1d	L1	3	23,4826	9,81276
23,5	9,8	23,499600000	23,483300000	9,828900000	9,812600000	0	1	0	3		10	y	3c	GL	4	23,4994	9,82904
23,5	9,8	23,515900000	23,499600000	9,845200000	9,828900000	1	0	0	4		10	y	4l	GL	4	23,5152	9,84504
23,5	9,9	23,532200000	23,515900000	9,861500000	9,845200000	0	1	1	4		10	y	4v	L2	4	23,5311	9,86152
23,5	9,9	23,548500000	23,532200000	9,877800000	9,861500000	1	0	0	4		10	y	4l	GL	4	23,5479	9,87767
27,2	13,5	27,167100000	-	13,496400000	-	0	-	-	1		-	y	1b	L1	3	27,1673	13,49615
27,2	13,5	27,183400000	27,167100000	13,512700000	13,496400000	1	0	0	3		10	y	3h	GL	4	27,1824	13,51259
27,2	13,5	27,199700000	27,183400000	13,529000000	13,512700000	0	1	0	4		10	n	4i	GL	4	27,1990	13,52912
27,2	13,5	27,216000000	27,199700000	13,545300000	13,529000000	1	0	0	4		10	n	4q	GL	4	27,2151	13,54520
27,2	13,6	27,232300000	27,216000000	13,561600000	13,545300000	0	1	1	4		10	n	4x	L2	4	27,2332	13,56150
27,2	13,6	27,248600000	27,232300000	13,577900000	13,561600000	1	0	0	4		10	n	4q	GL	4	27,2477	13,57773
30,4	16,7	30,361900000	-	16,691200000	-	0	-	-	1		-	y	1b	L1	3	30,3614	16,69102
30,4	16,7	30,378200000	30,361900000	16,707500000	16,691200000	1	0	0	3		10	y	3h	GL	4	30,3774	16,70756
30,4	16,7	30,394500000	30,378200000	16,723800000	16,707500000	0	1	0	4		10	y	4d	GL	4	30,3956	16,72363
30,5	16,8	30,508600000	30,394500000	16,837900000	16,723800000	1	0	0	4		70	n	4u	NO	2	NO CALC	NO CALC
30,5	16,9	30,524900000	30,508600000	16,854200000	16,837900000	0	1	1	2		10	n	2o	NO	2	NO CALC	NO CALC
30,5	16,9	30,541200000	30,524900000	16,870500000	16,854200000	1	0	0	2		10	n	2k	GL	4	30,5407	16,87073
30,6	16,9	30,557500000	30,541200000	16,886800000	16,870500000	0	1	0	4		10	n	4i	GL	4	30,5568	16,88702
33,4	19,7	33,361100000	-	19,690400000	-	0	-	-	1		-	y	1b	L1	3	33,3602	19,69017
33,4	19,7	33,377400000	33,361100000	19,706700000	19,690400000	1	0	0	3		10	y	3h	GL	4	33,3765	19,70689
33,4	19,7	33,393700000	33,377400000	19,723000000	19,706700000	0	1	0	4		10	y	4d	GL	4	33,3944	19,72313
33,5	19,8	33,507800000	33,393700000	19,837100000	19,723000000	1	0	0	4		70	y	4p	L1	3	33,5071	19,83722
33,5	19,9	33,524100000	33,507800000	19,853400000	19,837100000	0	1	1	3		10	y	3n	L1	3	33,5238	19,85313
33,5	19,9	33,540400000	33,524100000	19,869700000	19,853400000	1	0	0	3		10	y	3h	GL	4	33,5397	19,86999
33,6	19,9	33,556700000	33,540400000	19,886000000	19,869700000	0	1	0	4		10	y	4d	GL	4	33,5556	19,88571
36,1	22,5	36,132100000	-	22,461400000	-	1	-	-	1		-	y	1d	L1	3	36,1319	22,46135
36,1	22,5	36,148400000	36,132100000	22,477700000	22,461400000	0	1	0	3		10	y	3c	GL	4	36,1490	22,47788
36,2	22,5	36,164700000	36,148400000	22,494000000	22,477700000	1	0	0	4		10	y	4l	GL	4	36,1645	22,49431
36,3	22,6	36,278800000	36,164700000	22,608100000	22,494000000	0	1	0	4		70	y	4f	L1	3	36,2784	22,60821
36,3	22,6	36,295100000	36,278800000	22,624400000	22,608100000	1	0	1	3		10	n	3q	NO	2	NO CALC	NO CALC
36,3	22,6	36,311400000	36,295100000	22,640700000	22,624400000	0	1	0	2		10	n	2f	GL	4	36,3126	22,64107
36,3	22,7	36,327700000	36,311400000	22,657000000	22,640700000	1	0	0	4		10	n	4q	GL	4	36,3278	22,65682
38,7	25,1	38,723800000	-	25,053100000	-	1	-	-	1		-	y	1d	L1	3	38,7240	25,05280
38,7	25,1	38,740100000	38,723800000	25,069400000	25,053100000	0	1	0	3		10	n	3f	GL	4	38,7399	25,06919
38,8	25,1	38,756400000	38,740100000	25,085700000	25,069400000	1	0	0	4		10	n	4q	GL	4	38,7567	25,08535
38,9	25,2	38,870500000	38,756400000	25,199800000	25,085700000	0	1	0	4		70	n	4k	NO	2	NO CALC	NO CALC
38,9	25,2	38,886800000	38,870500000	25,216100000	25,199800000	1	0	1	2		10	y	2p	L1	3	38,8873	25,21601
38,9	25,2	38,903100000	38,886800000	25,232400000	25,216100000	0	1	0	3		10	y	3c	GL	4	38,9035	25,23266

lat sut	lon sut	lat last	lat prev	lon last	lon prev	tl	tp	tr	i		tim	o	op	cal	f	decoded lat	decoded lon
38,9	25,2	38,919400000	38,903100000	25,248700000	25,232400000	1	0	0	4		10	y	4l	GL	4	38,9199	25,24897
41,2	27,5	41,185100000	-	27,514400000	-	0	-	-	1		-	y	1b	L1	3	41,1844	27,51470
41,2	27,5	41,201400000	41,185100000	27,530700000	27,514400000	1	0	0	3		10	y	3h	GL	4	41,2006	27,53088
41,2	27,5	41,217700000	41,201400000	27,547000000	27,530700000	0	1	0	4		10	y	4d	GL	4	41,2186	27,54725
41,3	27,7	41,331800000	41,217700000	27,661100000	27,547000000	1	0	0	4		70	y	4p	L1	3	41,3312	27,66104
41,3	27,7	41,348100000	41,331800000	27,677400000	27,661100000	0	1	1	3		10	n	3o	NO	2	NO CALC	NO CALC
41,4	27,7	41,364400000	41,348100000	27,693700000	27,677400000	1	0	0	2		10	n	2k	GL	4	41,3639	27,69339
41,4	27,7	41,380700000	41,364400000	27,710000000	27,693700000	0	1	0	4		10	n	4i	GL	4	41,3797	27,71006
43,5	29,9	43,532300000	-	29,861600000	-	0	-	-	1		-	y	1b	L1	3	43,5311	29,86182
43,5	29,9	43,548600000	43,532300000	29,877900000	29,861600000	1	0	0	3		10	y	3h	GL	4	43,5491	29,87760
43,6	29,9	43,564900000	43,548600000	29,894200000	29,877900000	0	1	0	4		10	y	4d	GL	4	43,5653	29,89393
43,7	30,0	43,679000000	43,564900000	30,008300000	29,894200000	1	0	0	4		70	n	4u	NO	2	NO CALC	NO CALC
43,7	30,0	43,695300000	43,679000000	30,024600000	30,008300000	0	1	1	2		10	y	2n	L1	3	43,6947	30,02439
43,7	30,0	43,711600000	43,695300000	30,040900000	30,024600000	1	0	0	3		10	y	3h	GL	4	43,7124	30,04120
43,7	30,1	43,727900000	43,711600000	30,057200000	30,040900000	0	1	0	4		10	y	4d	GL	4	43,7289	30,05691
45,8	32,1	45,781700000	-	32,111000000	-	1	-	-	1		-	y	1d	L1	3	45,7821	32,11133
45,8	32,1	45,798000000	45,781700000	32,127300000	32,111000000	0	1	0	3		10	y	3c	GL	4	45,7973	32,12686
45,8	32,1	45,814300000	45,798000000	32,143600000	32,127300000	1	0	0	4		10	y	4l	GL	4	45,8148	32,14338
45,9	32,3	45,928400000	45,814300000	32,257700000	32,143600000	0	1	0	4		70	n	4k	NO	2	NO CALC	NO CALC
45,9	32,3	45,944700000	45,928400000	32,274000000	32,257700000	1	0	1	2		10	n	2q	NO	2	NO CALC	NO CALC
46,0	32,3	45,961000000	45,944700000	32,290300000	32,274000000	0	1	0	2		10	n	2f	GL	4	45,9609	32,28987
46,0	32,3	45,977300000	45,961000000	32,306600000	32,290300000	1	0	0	4		10	n	4q	GL	4	45,9780	32,30655
47,9	34,3	47,949600000	-	34,278900000	-	1	-	-	1		-	y	1d	L1	3	47,9498	34,27848
48,0	34,3	47,965900000	47,949600000	34,295200000	34,278900000	0	1	0	3		10	n	3f	GL	4	47,9658	34,29500
48,0	34,3	47,982200000	47,965900000	34,311500000	34,295200000	1	0	0	4		10	n	4q	GL	4	47,9824	34,31192
48,1	34,4	48,096300000	47,982200000	34,425600000	34,311500000	0	1	0	4		70	y	4f	L1	3	48,0952	34,42593
48,1	34,4	48,112600000	48,096300000	34,441900000	34,425600000	1	0	1	3		10	y	3p	L1	3	48,1130	34,44224
48,1	34,5	48,128900000	48,112600000	34,458200000	34,441900000	0	1	0	3		10	y	3c	GL	4	48,1294	34,45809
48,1	34,5	48,145200000	48,128900000	34,474500000	34,458200000	1	0	0	4		10	y	4l	GL	4	48,1457	34,47421

7.4.3.1.5 Tables of values for use in content checking test cases

Table 7.15: Values for use in the test "Sync_Format"

SYNC_BURST_APPIN_PARAMETERS(x)					**SYNC_BURST_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)
a/d	Random transmission or reserved transmission in a slot selected by this station.	Delayed burst transmission in a slot selected by a peer station.	Random transmission or reserved transmission in a slot selected by this station.	Delayed burst transmission in a slot selected by a peer station.	0	1	0	1
rid	Reservation type is a periodic broadcast reservation.	Reservation type is a periodic broadcast reservation.	Reservation type is a periodic broadcast reservation.	Reservation type is a combined periodic broadcast and incremental broadcast reservation.	1	1	1	1
ver	Version 0	Version 0	Version 0	Version 0	000	000	000	000
s	add_B	add_B	add_B	add_B	add_B	add_B	add_B	add_B
TCP change flag (tqc)	Encoded as zero if the a/d flag indicates a directed sync burst, otherwise encoded as one.	Encoded as zero if the a/d flag indicates a directed sync burst, otherwise encoded as one.	Encoded as zero if the a/d flag indicates a directed sync burst, otherwise encoded as one.	Encoded as zero if the a/d flag indicates a directed sync burst, otherwise encoded as one.	1	0	1	0
baro/geo altitude (b/g)	Geometric base altitude	Barometric base altitude	Geometric base altitude	Barometric base altitude	1	0	1	0
CPR format even/odd (cprf)	As encoded in test CPR_Encode	As encoded in test CPR_Encode	As encoded in test CPR_Encode	As encoded in test CPR_Encode	As encoded in test CPR_Encode	As encoded in test CPR_Encode	As encoded in test CPR_Encode	As encoded in test CPR_Encode
Navigation integrity channel (nic)	Horizontal and vertical containment radius < 7,5 m	Horizontal and vertical containment radius < 0,2 NM	Horizontal and vertical containment radius > 20 NM	Horizontal and vertical containment radius < 25 m	11	7	0	10
latitude (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
base altitude (balt)	8 025 feet	-1 300 feet	130 000 feet	0 feet	936	2	4 071	132
longitude (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

SYNC_BURST_APPIN_PARAMETERS(x)					**SYNC_BURST_RF_OUT_PARAMETERS(x)**			
Time figure of merit (tfom)	Primary non-certified	Primary non-certified	Primary non-certified	Secondary	1	1	1	2
data age (da)	latency 250 ms	latency 50 ms	latency 3 500 ms	latency 0 ms	2	0	14	0
Information field (in)	No information field present	No information field present	No information field present	No information field present	F hex	F hex	F hex	F hex

Table 7.16: Values for use in the test "Sync_Format_Rec"

SYNC_BURST_RF_PARAMETERS(x)					**SYNC_BURST_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)
a/d	0	1	0	1	Random transmission or reserved transmission in a slot selected by this station	Delayed burst transmission in a slot selected by a peer station.	Random transmission or reserved transmission in a slot selected by this station.	Delayed burst transmission in a slot selected by a peer station.
rid	1	1	1	1	Reservation type is a periodic broadcast reservation.	Reservation type is a periodic broadcast reservation.	Reservation type is a periodic broadcast reservation.	Reservation type is a combined periodic broadcast and incremental broadcast reservation.
ver	000	000	000	000	Version 0	Version 0	Version 0	Version 0
s	add_B	add_B	add_B	add_B	add_B	add_B	add_B	add_B
TCP change flag (tqc)	1	0	1	0	Encoded as zero if the a/d flag indicates a directed sync burst, otherwise encoded as one.	Encoded as zero if the a/d flag indicates a directed sync burst, otherwise encoded as one.	Encoded as zero if the a/d flag indicates a directed sync burst, otherwise encoded as one.	Encoded as zero if the a/d flag indicates a directed sync burst, otherwise encoded as one.
baro/geo altitude (b/g)	1	0	1	0	Geometric base altitude	Barometric base altitude	Geometric base altitude	Barometric base altitude
CPR format even/odd (cprf)	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
Navigation integrity channel (nic)	11	7	0	10	Horizontal and vertical containment radius < 7,5 m	Horizontal and vertical containment radius < 0,2 NM	Horizontal and vertical containment radius > 20 NM	Horizontal and vertical containment radius < 25 m
latitude (lat)	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

SYNC_BURST_RF_PARAMETERS(x)					SYNC_BURST_APPOUT_PARAMETERS(x)			
base altitude (balt)	936	2	4 071	132	8 025 feet	-1 300 feet	130 000 feet	0 feet
longitude (lon)	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
Time figure of merit (tfom)	0	1	0	2	Primary certified	Primary non-certified	Primary certified	Secondary
data age (da)	2	0	14	0	latency 250 ms	latency 50 ms	latency 3 500 ms	latency 0 ms
Information field (in)	F hex	F hex	F hex	F hex	No information field present	No information field present	No information field present	No information field present

7.4.3.1.6 VDL4 burst formats

In the following definitions, the function int(x) shall be taken to mean the largest integer less than or equal to x.

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.

Table 7.17: SYNC_BURST_a (Sa): Information field all "0"s, occupies one slot, lat and lon specified

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	x
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	x	x	x	x	x	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	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
in	15	0	0	0	0	0	0	0	0
in	16	0	0	0	0	0	0	0	0
in	17	0	0	0	0	0	0	0	0
in, 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 7.18: SYNC_BURST_b (Sb): Information field contains "0"s, 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	x
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	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	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
in	15	0	0	0	0	0	0	0	0
in	16	0	0	0	0	0	0	0	0
in	17	0	0	0	0	0	0	0	0
in, 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 7.19: SYNC_BURST_c (Sc): Occupies one slot, autonomous 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	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, lat	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	x	x	x	x
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

NOTE: The above format includes the a/d flag encoded as a zero in bit 1 of octet 1.

Table 7.20: SYNC_BURST_d(*k*) (Sd(*k*)): Information field contains "0"s, occupies exactly k 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	x
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	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	1	1
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
		Insert int(31,5 x (*k* - 1)) repeat rows							
in	15 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	16 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	17 + int(31,5) x (*k* - 1)	0	0	0	0	0	0	0	0
in, pt	18 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	pt_2	pt_1
po	19 + int(31,5 x (*k* - 1))	po_8	po_7	po_6	po_5	po_4	po_3	po_2	po_1
c	20 + int(31,5 x (*k* - 1))	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21 + int(31,5 x (*k* - 1))	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 7.21: SYNC_BURST_e (Se): Information field all "0"s, exceeds slot boundary by 5 octets

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	x
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	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	1	1
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
in	15	0	0	0	0	0	0	0	0
in	16	0	0	0	0	0	0	0	0
in	17	0	0	0	0	0	0	0	0
in	18	0	0	0	0	0	0	0	0
in	19	0	0	0	0	0	0	0	0
in	20	0	0	0	0	0	0	0	0
in	21	0	0	0	0	0	0	0	0
in	22	0	0	0	0	0	0	0	0
in, pt	23	0	0	0	0	0	0	pt_2	pt_1
po	24	po_8	po_7	po_6	po_5	po_4	po_3	po_2	po_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 7.22: SYNC_BURST_f(*k*) (Sf(*k*)): Non-zero ver. Information field all "0"s, occupies k 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	1	1	x
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	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	1	1
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
			Insert int(31,5 x (*k* - 1)) repeat rows						
in	15 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	16 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	17 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in, pt	18 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	pt_{2}	pt_{1}
po	19 + int(31,5 x (*k* - 1))	po_{8}	po_{7}	po_{6}	po_{5}	po_{4}	po_{3}	po_{2}	po_{1}
c	20 + int(31,5 x (*k* - 1))	c_{9}	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21 + int(31,5 x (*k* - 1))	c_{1}	c_{2}	c_{3}	c_{4}	c_{5}	c_{6}	c_{7}	c_{8}
flag	-	0	1	1	1	1	1	1	0

Table 7.23: SYNC_BURST_k(*k*) (Sd(*k*)): Information field all "0"s, occupies k slots, lat and lon specified

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	x
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	x	x	x	x	x	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	1
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
		Insert int(31,5 x (*k* - 1)) repeat rows							
in	15 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	16 + int(31,5 x (*k*- 1))	0	0	0	0	0	0	0	0
in	17 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in, pt	18 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	pt_2	pt_1
po	19 + int(31,5 x (*k* - 1))	po_8	po_7	po_6	po_5	po_4	po_3	po_2	po_1
c	20 + int(31,5 x (*k* - 1))	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21 + int(31,5 x (*k* - 1))	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.24: SYNC_BURST_l (Sl): 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	x
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	x	x	x	x
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	x	x
po	19	x	x	x	x	x	x	x	x
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 7.25: SYNC_BURST_m (Sm): Directed sync burst with response reservation, directed 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
nic, 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	x	x	x	x
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
d	16	d_{24}	d_{23}	d_{22}	d_{21}	d_{20}	d_{19}	d_{18}	d_{17}
d	17	d_{16}	d_{15}	d_{14}	d_{13}	d_{12}	d_{11}	d_{10}	d_9
d	18	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1
erid, d	19	0	0	0	0	0	d_{27}	d_{26}	d_{25}
c	20	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.26: SYNC_BURST_n (Sn): Information field contains "0"s, extends past one slot boundary by 2 octets

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	x
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	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	1	1
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
in	15	0	0	0	0	0	0	0	0
in	16	0	0	0	0	0	0	0	0
in	17	0	0	0	0	0	0	0	0
in	18	0	0	0	0	0	0	0	0
in	19	0	0	0	0	0	0	0	0
in, pt	20	0	0	0	0	0	0	pt_2	pt_1
po	21	po_8	po_7	po_6	po_5	po_4	po_3	po_2	po_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 7.27: RAND_ACC_DATA_a (Ra): Information field contains "0"s, 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
in, mi	5	0	0	0	0	0	1	0	1
in	6	0	0	0	0	0	0	0	0
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	0	0	0	0	0	0	0
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
in	15	0	0	0	0	0	0	0	0
in	16	0	0	0	0	0	0	0	0
in	17	0	0	0	0	0	0	0	0
in	18	0	0	0	0	0	0	0	0
erid, in	19	0	0	0	0	0	0	0	0
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 7.28: BURST_UNREC_a (Ba): Information field contains "0"s, 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	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	0	0	0	0	1	0	1
in	6	0	0	0	0	0	0	0	0
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	0	0	0	0	0	0	0
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
in	15	0	0	0	0	0	0	0	0
in	16	0	0	0	0	0	0	0	0
in	17	0	0	0	0	0	0	0	0
in	18	0	0	0	0	0	0	1	1
erid, io	19	0	0	1	1	1	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 7.29: UNI_BURST_a (Ua): Information field all "0"s, 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	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	0	0	0	0	1	0	1
in	6	0	0	0	0	0	0	0	0
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	0	0	0	0	0	0	0
in	12	0	0	0	0	0	0	0	0
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
ro, sdf, d	16	ro_{12}	ro_{11}	ro_{10}	ro_9	sdf	d_{27}	d_{26}	d_{25}
ro	17	ro_8	ro_7	ro_6	ro_5	ro_4	ro_3	ro_2	ro_1
lg	18	lg_8	lg_7	lg_6	lg_5	lg_4	lg_3	lg_2	lg_1
erid, pr	19	0	0	1	0	pr_4	pr_3	pr_2	pr_1
c	20	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.30: UNI_BURST_b (Ub): Invalid message ID, Information field all "0"s, 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	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	1	0	1	0	1
in	6	0	0	0	0	0	0	0	0
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	0	0	0	0	0	0	0
in	12	0	0	0	0	0	0	0	0
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
ro, sdf, d	16	ro_{12}	ro_{11}	ro_{10}	ro_9	sdf	d_{27}	d_{26}	d_{25}
ro	17	ro_8	ro_7	ro_6	ro_5	ro_4	ro_3	ro_2	ro_1
lg	18	lg_8	lg_7	lg_6	lg_5	lg_4	lg_3	lg_2	lg_1
erid, pr	19	0	0	1	0	pr_4	pr_3	pr_2	pr_1
c	20	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.31: UNI_BURST_c (Uc): For source to broadcast, Information field all "0"s, 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	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	0	0	0	0	1	0	1
in	6	0	0	0	0	0	0	0	0
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	0	0	0	0	0	0	0
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
in	15	0	0	0	0	0	0	0	0
ro, sdf, d	16	ro_{12}	ro_{11}	ro_{10}	ro_9	1	1	1	1
ro	17	ro_8	ro_7	ro_6	ro_5	ro_4	ro_3	ro_2	ro_1
lg	18	lg_8	lg_7	lg_6	lg_5	lg_4	lg_3	lg_2	lg_1
erid, pr	19	0	0	1	0	pr_4	pr_3	pr_2	pr_1
c	20	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.32: UNI_BURST_d (Ud): With general request, Information field all "0"s, 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	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_5	r-mi_4	r-mi_3	r-mi_2	r-mi_1	0	0	1
res, r-mi	6	0	0	0	0	0	0	r-mi_7	r-mi_6
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	0	0	0	0	0	0	0
in	12	0	0	0	0	0	0	0	0
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
ro, sdf, d	16	ro_{12}	ro_{11}	ro_{10}	ro_9	sdf	d_{27}	d_{26}	d_{25}
ro	17	ro_8	ro_7	ro_6	ro_5	ro_4	ro_3	ro_2	ro_1
lg	18	lg_8	lg_7	lg_6	lg_5	lg_4	lg_3	lg_2	lg_1
erid, pr	19	0	0	1	0	pr_4	pr_3	pr_2	pr_1
c	20	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.33: INCREM_BURST_a (Ia): Information field contains "0"s, 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	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	0	0	0	0	1	0	1
in	6	0	0	0	0	0	0	0	0
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	0	0	0	0	0	0	0
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
in	15	0	0	0	0	0	0	0	0
in	16	0	0	0	0	0	0	0	0
in	17	0	0	0	0	0	0	0	0
in	18	0	0	0	0	0	0	io_8	io_7
erid, io	19	1	0	io_6	io_5	io_4	io_3	io_2	io_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 7.34: INCREM_BURST_b(k) (Ib(k)): Information field contains "0"s, occupies exactly k 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	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	0	0	0	0	1	0	1
in	6	0	0	0	0	0	0	0	0
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	0	0	0	0	0	0	0
		Insert int(31,5 x (*k* - 1)) repeat rows							
in	12 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	13 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	14 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	15 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	16 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	17 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	0	0
in	18 + int(31,5 x (*k* - 1))	0	0	0	0	0	0	io_8	io_7
erid, io	19 + int(31,5 x (*k* - 1))	1	0	io_6	io_5	io_4	io_3	io_2	io_1
c	20 + int(31,5 x (*k* - 1))	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	21 + int(31,5 x (*k* - 1))	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.35: INCREM_BURST_c (Ic): Invalid message ID. Information field all "0"s, 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	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	1	0	1	0	1
in	6	0	0	0	0	0	0	0	0
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	0	0	0	0	0	0	0
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
in	15	0	0	0	0	0	0	0	0
in	16	0	0	0	0	0	0	0	0
in	17	0	0	0	0	0	0	0	0
in	18	0	0	0	0	0	0	io_8	io_7
erid, io	19	1	0	io_6	io_5	io_4	io_3	io_2	io_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 7.36: NULL_RES_a (Na): Information field contains "0"s, 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	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	0	0	0	0	1	0	1
in	6	0	0	0	0	0	0	0	0
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	0	0	0	0	0	0	0
in	12	0	0	0	0	0	0	0	0
in	13	0	0	0	0	0	0	0	0
in	14	0	0	0	0	0	0	0	0
in	15	0	0	0	0	0	0	0	0
in	16	0	0	0	0	0	0	0	0
in	17	0	0	0	0	0	0	0	0
in	18	0	0	0	0	0	0	0	0
rd	19	0	0	0	0	0	0	0	0
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 7.37: INF_TRANS_a (ITa): Information field contains "0"s, 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	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	0	0	0	0	1	0	1
in	6	0	0	0	0	0	0	0	0
in	7	0	0	0	0	0	0	0	0
in	8	0	0	0	0	0	0	0	0
in	9	0	0	0	0	0	0	0	0
in	10	0	0	0	0	0	0	0	0
in	11	0	ao_7	ao_6	ao_5	ao_4	ao_3	ao_2	ao_1
lg	12	lg_8	lg_7	lg_6	lg_5	lg_4	lg_3	lg_2	lg_1
ro	13	ro_8	ro_7	ro_6	ro_5	ro_4	ro_3	ro_2	ro_1
ro, f	14	ro_{12}	ro_{11}	ro_{10}	ro_9	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
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, sdf, d	19	0	1	0	1	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 7.38: DIR_REQ_a (Da): Contains general request, Information field contains "0"s, 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	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_5	r-mi_4	r-mi_3	r-mi_2	r-mi_1	0	0	1
res, r-mi	6	0	0	0	0	0	0	r-mi_7	r-mi_6
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	$rcvr_2$	$rcvr_1$	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, 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 7.39: DIR_SYNC_BURST_a (DSa): Directed sync burst, 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	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
nic, 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	x	x	x	x
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 7.40: GEN_RESP_a (GRa): General response burst with response reservation

Description	Octet	Bit number							
		8	**7**	**6**	**5**	**4**	**3**	**2**	**1**
flag	-	0	1	1	1	1	1	1	0
s, ver, rid, a/d	1	s_{27}	s_{26}	s_{25}	0	0	0	0	1
s	2	s_{24}	s_{23}	s_{22}	s_{21}	s_{20}	s_{19}	s_{18}	s_{17}
s	3	s_{16}	s_{15}	s_{14}	s_{13}	s_{12}	s_{11}	s_{10}	s_9
s	4	s_8	s_7	s_6	s_5	s_4	s_3	s_2	s_1
ok, mi	5	ok	1	1	1	0	1	0	1
res, r-mi	6	0	$r\text{-}mi_7$	$r\text{-}mi_6$	$r\text{-}mi_5$	$r\text{-}mi_4$	$r\text{-}mi_3$	$r\text{-}mi_2$	$r\text{-}mi_1$
bd	7	bd_8	bd_7	bd_6	bd_5	bd_4	bd_3	bd_2	bd_1
err	8	err_8	err_7	err_6	err_5	err_4	err_3	err_2	err_1
d	9	d_{24}	d_{23}	d_{22}	d_{21}	d_{20}	d_{19}	d_{18}	d_{17}
d	10	d_{16}	d_{15}	d_{14}	d_{13}	d_{12}	d_{11}	d_{10}	d_9
d	11	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1
erid, d	132	0	0	0	0	0	d_{27}	d_{26}	d_{25}
c	13	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	14	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.41: 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 7.42: 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	0	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 7.43: PLEA_RESP_b (PRb): 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	0	0	0	0	1	0	1
res, a	6	0	0	0	0	$a_{5,12}$	$a_{5,11}$	$a_{5,10}$	$a_{5,9}$
a	7	$a_{5,8}$	$a_{5,7}$	$a_{5,6}$	$a_{5,5}$	$a_{5,4}$	$a_{5,3}$	$a_{5,2}$	$a_{5,1}$
a	8	$a_{4,8}$	$a_{4,7}$	$a_{4,6}$	$a_{4,5}$	$a_{4,4}$	$a_{4,3}$	$a_{4,2}$	$a_{4,1}$
a	9	$a_{4,12}$	$a_{4,11}$	$a_{4,10}$	$a_{4,9}$	$a_{3,12}$	$a_{3,11}$	$a_{3,10}$	$a_{3,9}$
a	10	$a_{3,8}$	$a_{3,7}$	$a_{3,6}$	$a_{3,5}$	$a_{3,4}$	$a_{3,3}$	$a_{3,2}$	$a_{3,1}$
a	11	$a_{2,8}$	$a_{2,7}$	$a_{2,6}$	$a_{2,5}$	$a_{2,4}$	$a_{2,3}$	$a_{2,2}$	$a_{2,1}$
a	12	$a_{2,12}$	$a_{2,11}$	$a_{2,10}$	$a_{2,9}$	$a_{1,12}$	$a_{1,11}$	$a_{1,10}$	$a_{1,9}$
a	13	$a_{1,8}$	$a_{1,7}$	$a_{1,6}$	$a_{1,5}$	$a_{1,4}$	$a_{1,3}$	$a_{1,2}$	$a_{1,1}$
res, off	14	0	0	off_9	off_8	off_7	off_6	off_5	off_4
off, pr_flag, nr	15	off_3	off_2	off_1	1	1	1	1	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 7.44: SUPER_BLOCK_a (SUa): Superframe block 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
in, mi	5	0	0	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
blg, d	9	blg_5	blg_4	blg_3	blg_2	blg_1	d_{27}	d_{26}	d_{25}
roff	10	$roff_8$	$roff_7$	$roff_6$	$roff_5$	$roff_4$	$roff_3$	$roff_2$	$roff_1$
br	11	0	0	0	0	br_4	br_3	br_2	br_1
bs	12	bs_8	bs_7	bs_6	bs_5	bs_4	bs_3	bs_2	bs_1
bo	13	bo_8	bo_7	bo_6	bo_5	bo_4	bo_3	bo_2	bo_1
bt	14	0	0	0	1	0	0	bt_2	bt_1
c	15	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	16	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.45: RE_BROADCAST_a (RBa): Superframe block re-broadcast 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
in, mi	5	0	0	0	0	0	1	0	1
blg, d	9	blg_5	blg_4	blg_3	blg_2	blg_1	0	0	0
roff	10	$roff_8$	$roff_7$	$roff_6$	$roff_5$	$roff_4$	$roff_3$	$roff_2$	$roff_1$
br	11	0	0	0	0	br_4	br_3	br_2	br_1
bs	12	bs_8	bs_7	bs_6	bs_5	bs_4	bs_3	bs_2	bs_1
bo	13	bo_8	bo_7	bo_6	bo_5	bo_4	bo_3	bo_2	bo_1
bt	14	0	0	0	1	0	0	bt_2	bt_1
c	15	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	16	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.46: SECOND_BLOCK_a (SCa): Second frame block 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
in, mi	5	0	0	0	0	0	1	0	1
vt, sz	6	vt_6	vt_5	vt_4	vt_3	vt_2	vt_1	sz_5	sz_4
erid, sz	7	0	0	0	1	1	sz_3	sz_2	sz_1
c	8	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	9	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: UCTRL_VS (Cv): UCTRL burst for VS1, VS2, VS4, and VS5 parameters.

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
ucid, ucd	5	0	0	0	0	0	0	1	1
VSS parameter id	6	0	1	0	0	0	0	0	1
parameter length	7	0	0	0	0	0	0	1	1
VS5, VS1	8	$VS5_4$	$VS5_3$	$VS5_2$	$VS5_1$	$VS1_4$	$VS1_3$	$VS1_2$	$VS1_1$
VS2	9	0	0	$VS2_6$	$VS2_5$	$VS2_4$	$VS2_3$	$VS2_2$	$VS2_1$
VS4	10	0	$VS4_7$	$VS4_6$	$VS4_5$	$VS4_4$	$VS4_3$	$VS4_2$	$VS4_1$
res	11	0	0	0	0	0	0	0	0
null reservation	12	0	0	0	0	0	0	0	0
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 7.48: UCTRL_CG (Cg): UCTRL burst for CG1 parameters

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
ucid, ucd	5	0	0	0	0	0	0	1	1
CG1 parameter id	6	0	1	0	0	0	1	0	0
parameter length	7	0	0	0	0	0	1	1	1
CG1_plea	8	p_8	p_7	p_6	p_5	p_4	p_3	p_2	p_1
CG1_range	9	r_8	r_7	r_6	r_5	r_4	r_3	r_2	r_1
TL5	10	t_8	t_7	t_6	t_5	t_4	t_3	t_2	t_1
CG1_limit	11	l_{16}	l_{15}	l_{14}	l_{13}	l_{12}	l_{11}	l_{10}	l_9
	12	l_8	l_7	l_6	l_5	l_4	l_3	l_2	l_1
CG1_inc	13	i_8	i_7	i_6	i_5	i_4	i_3	i_2	i_1
1/CG1_decay	14	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1
res	15	0	0	0	0	0	0	0	0
null reservation	16	0	0	0	0	0	0	0	0
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 7.49: UCTRL_M2 (Cm): UCTRL burst for M2 parameters

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
ucid, ucd	5	0	0	0	0	0	0	1	1
m2 parameter id	6	0	1	0	0	0	0	1	1
parameter length	7	0	0	0	0	0	0	1	1
M2inc	8	i_8	i_7	i_6	i_5	i_4	i_3	i_2	i_1
M2limit	9	l_{16}	l_{15}	l_{14}	l_{13}	l_{12}	l_{11}	l_{10}	l_9
	10	l_8	l_7	l_6	l_5	l_4	l_3	l_2	l_1
res	11	0	0	0	0	0	0	0	0
null reservation	12	0	0	0	0	0	0	0	0
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 7.50: UCTRL_Q (Cq): UCTRL burst for QoS parameters

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
ucid, ucd	5	0	0	0	0	0	0	1	1
Q4 parameter id	6	0	1	0	0	0	0	1	0
parameter length	7	0	0	0	0	0	1	1	1
Q1max, Q1min	8	$Q1max_1$	$Q1max_1$	$Q1max_1$	$Q1max_1$	$Q1min_1$	$Q1min_1$	$Q1min_1$	$Q1min_1$
Q2a	8	$Q2a_8$	$Q2a_7$	$Q2a_6$	$Q2a_5$	$Q2a_4$	$Q2a_3$	$Q2a_2$	$Q2a_1$
Q2b	10	$Q2b_8$	$Q2b_7$	$Q2b_6$	$Q2b_5$	$Q2b_4$	$Q2b_3$	$Q2b_2$	$Q2b_1$
Q2c	11	$Q2c_8$	$Q2c_7$	$Q2c_6$	$Q2c_5$	$Q2c_4$	$Q2c_3$	$Q2c_2$	$Q2c_1$
Q2d	12	$Q2d_8$	$Q2d_7$	$Q2d_6$	$Q2d_5$	$Q2d_4$	$Q2d_3$	$Q2d_2$	$Q2d_1$
Q2a,b,c,d	13	$Q2d_{10}$	$Q2d_9$	$Q2c_{10}$	$Q2c_9$	$Q2b_{10}$	$Q2b_9$	$Q2a_{10}$	$Q2a_9$
Q4	14	0	0	0	$Q4_5$	$Q4_4$	$Q4_3$	$Q4_2$	$Q4_1$
res	15	0	0	0	0	0	0	0	0
null reservation	16	0	0	0	0	0	0	0	0
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 7.51: BND_DELAYED_a (BDa): Contains BND reservation, 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
res, nd	6	0	0	0	0	0	0	nd_5	nd_4
erid, nd	7	0	0	0	0	1	nd_3	nd_2	nd_1
c	8	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	9	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: BND_LONG_b (BDb): Contains BND reservation, 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}	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	0	0	0	0	1	0	1
res	6	0	0	0	0	0	0	0	0
res	7	0	0	0	0	0	0	0	0
res	8	0	0	0	0	0	0	0	0
res	9	0	0	0	0	0	0	0	0
res	10	0	0	0	0	0	0	0	0
res	11	0	0	0	0	0	0	0	0
res	12	0	0	0	0	0	0	0	0
res	13	0	0	0	0	0	0	0	0
res	14	0	0	0	0	0	0	0	0
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, nd	18	0	0	0	0	0	0	nd_5	nd_4
erid, nd	19	0	0	0	0	1	nd_3	nd_2	nd_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: CTRL_RTS_a (CRa): CTRL_RTS DLPDU with unicast reservation

Description	Octet	Bit number							
		8	**7**	**6**	**5**	**4**	**3**	**2**	**1**
flag	-	0	1	1	1	1	1	1	0
a/d, rid, ver	1	s_{27}	s_{26}	s_{25}	ver_3	ver_2	ver_1	Rid	a/d
s	2	s_{24}	s_{23}	s_{22}	s_{21}	s_{20}	s_{19}	s_{18}	s_{17}
s	3	s_{16}	s_{15}	s_{14}	s_{13}	s_{12}	s_{11}	s_{10}	s_9
s	4	s_8	s_7	s_6	s_5	s_4	s_3	s_2	s_1
burst id,	5	0	0	0	0	1	1	0	1
lg, T, IB	6	0	T	IB	0	1	0	0	1
pr	7	pr_4	pr_3	pr_2	pr_1	lg_4	lg_3	lg_2	lg_1
d	8	d_{24}	d_{23}	d_{22}	d_{21}	d_{20}	d_{19}	d_{18}	d_{17}
d	9	d_{16}	d_{15}	d_{14}	d_{13}	d_{12}	d_{11}	d_{10}	d_9
d	10	d_8	d_7	d_6	d_5	d_4	d_3	d_2	d_1
sdf	11	ro_{12}	ro_{11}	ro_{10}	ro_9	sdf	d_{27}	d_{26}	d_{25}
ro	12	ro_8	ro_7	ro_6	ro_5	ro_4	ro_3	ro_2	ro_1
lg	13	lg_8	lg_7	lg_6	lg_5	lg_4	lg_3	lg_2	lg_1
erid, pr	14	0	0	1	0	pr_4	pr_3	pr_2	pr_1
c	13	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	14	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
flag	-	0	1	1	1	1	1	1	0

Table 7.54: UINFO_a (UIa): UINFO DLPDU with response reservation with address type field 7

Description	Octet	Bit number							
		8	**7**	**6**	**5**	**4**	**3**	**2**	**1**
flag	-	0	1	1	1	1	1	1	0
a/d, rid, ver	1	s_{27}	s_{26}	s_{25}	ver_3	ver_2	ver_1	rid	a/d
s	2	s_{24}	s_{23}	s_{22}	s_{21}	s_{20}	s_{19}	s_{18}	s_{17}
s	3	s_{16}	s_{15}	s_{14}	s_{13}	s_{12}	s_{11}	s_{10}	s_9
s	4	s_8	s_7	s_6	s_5	s_4	s_3	s_2	s_1
burst id	5	$ud1_5$	$ud1_4$	$ud1_3$	$ud1_2$	$ud1_1$	1	1	1
inf	7	information field							
erid, d	8	0	0	0	0	0	1	1	1
c	9	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	10	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: UINFO_b (UIb): Invalid UINFO DLPDU with response reservation with address type field equal to 7 (bit 8 of octet 5 is not present)

Description	Octet	Bit number							
		8	**7**	**6**	**5**	**4**	**3**	**2**	**1**
flag	-	0	1	1	1	1	1	1	0
a/d, rid, ver	1	s_{27}	s_{26}	s_{25}	ver_3	ver_2	ver_1	Rid	a/d
s	2	s_{24}	s_{23}	s_{22}	s_{21}	s_{20}	s_{19}	s_{18}	s_{17}
s	3	s_{16}	s_{15}	s_{14}	s_{13}	s_{12}	s_{11}	s_{10}	s_9
s	4	s_8	s_7	s_6	s_5	s_4	s_3	s_2	s_1
burst id	5	No bit	$ud1_4$	$ud1_3$	$ud1_2$	$ud1_1$	1	1	1
erid, d	8	0	0	0	0	0	1	1	1
c	9	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	10	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: UCTRL_a (UCa): UCTRL DLPDU with response reservation with address type to 7

Description	Octet	Bit number							
		8	**7**	**6**	**5**	**4**	**3**	**2**	**1**
flag	-	0	1	1	1	1	1	1	0
a/d, rid, ver	1	s_{27}	s_{26}	s_{25}	ver_3	ver_2	ver_1	rid	a/d
s	2	s_{24}	s_{23}	s_{22}	s_{21}	s_{20}	s_{19}	s_{18}	s_{17}
s	3	s_{16}	s_{15}	s_{14}	s_{13}	s_{12}	s_{11}	s_{10}	s_9
s	4	s_8	s_7	s_6	s_5	s_4	s_3	s_2	s_1
ucid, ucd	5	$ucid_5$	$ucid_4$	$ucid_3$	$ucid_2$	$ucid_1$	0	1	1
inf	7	information field							
erid, d	8	0	0	0	0	0	1	1	1
c	9	c_9	c_{10}	c_{11}	c_{12}	c_{13}	c_{14}	c_{15}	c_{16}
c	10	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 must be brought into the defined idle state before the performance of the individual test cases. Each test case starts in this state and leaves the equipment in that state after completion. The idle state is the state which the equipment enters after successful completion of the power up sequence. To perform several test cases in sequence the power on macro M_POWER_UP must only be executed at the beginning.

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

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

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

7.4.3.2.1 Test case macros

The following macros are used in several test cases.

DATA_a(*m*) (Da(*m*)): Definition: Fill m bits of data with "0"s followed by "1"s. Bit number 1 is "0".

m odd

bit_m	bit_{m-1}	bit_{m-2}		bit_4	bit_3	bit_2	bit_1
0	1	0		1	0	1	0

m even

bit_m	bit_{m-1}	bit_{m-2}		bit_4	bit_3	bit_2	bit_1
1	0	1		1	0	1	0

Macro Name: M_POWER_UP				VDL4 ground station power up.		
Parameters:						
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
macro	1	do		Switch on VDL4 ground station		
	2	verify	Self test	Successful VDL4 ground station BITE self test		Verify that the VDL4 ground station has successfully passed BITE power-up test.
	3	wait		3 minutes		Wait for ground station to acquire reservation table and default into idle state.
	4	send	Position	Input test station's position		Inform station under test of its own position.
	5	record		*add_A*:= address of station under test		
	6	send	VSS	SET PARAMETERS (V66:= 0)		Set the second frame block reservation to 0.
Comments: This macro prepares the VDL4 ground station for testing. It brings the VDL4 ground station into the defined idle state.						

Macro Name: M_RAND_ACC_SU (sf)				Establish a queue of random access transmissions over a number of superframes.		
Parameters: (sf = number of superframes to transmit over)						
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
macro	1	repx		*n*:= 0; *sf*:= no. of superframes to transmit over		Maintains transmissions over sf superframes.
	2	queue	VSS	DATA_a(*m*)	Da(*m*)	Send packets of data (labelled DATA_a) to the station under test for subsequent transmission by the random access protocol. Identify packets with repeating 10101010 bit sequence over *m* bits.
	3	until		*n* = *sf* x M1; *n*:= *n* + 1		Send M1 x *sf* random access transmissions.
Comments: Establishes a queue of random access transmissions over a defined number of superframes. Each random access is transmitted as a discrete burst, requiring the station under test to verify the state of the channel at the slot boundary prior to transmission. Flow control must be implemented at the VSS User PCO to ensure that the station under test is not flooded. This macro tests which slots the station considers occupied. It is acceptable for implementations to use other means to provide this information (e.g. with a command on the VSS PCO).						

Macro Name: M_RAND_ACC_SL (slots)	Establish a queue of random access transmissions over a number of slots.					
Parameters: (slots = number of slots to transmit over)						
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
macro	1	repx		*n*:= 0; *slots*:= no. of slots to transmit over		Maintains transmissions over sf superframes.
	2	queue	VSS	DATA_a(*m*)	Da(*m*)	Send packets of data (labelled DATA_a) to the station under test for subsequent transmission by the random access protocol. Identify packets with repeating 10101010 bit sequence over *m* bits.
	3	until		*n* = *slots*; *n*:= *n* + 1		Send *slots* random access transmissions.
Comments: Establishes a queue of random access transmissions over a defined number of superframes. Each random access is transmitted as a discrete burst, requiring the station under test to verify the state of the channel at the slot boundary prior to transmission. Flow control must be implemented at the VSS User PCO to ensure that the station under test is not flooded. This macro tests which slots the station considers occupied. It is acceptable for implementations to use other means to provide this information (e.g., with a command on the VSS PCO).						

Macro Name: M_ASSIGN_SLOTS (pos1, pos2)	Chooses two random slot positions.					
Parameters: (pos1 = variable to which first slot position is assigned, pos2 = variable to which second slot position is assigned)						
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
macro	1	record		*pos1*:= 64 + 4 x RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	2	repx				
	3			*pos2*:= 64 + 4 x RAND(0, 5)		Choose another slot position within the candidate range.
	4	until		*pos2* ≠ *pos1*		Ensure *random_position_2* differs from *random_position_1*.
	5	do		**IF** *pos2* < *pos1* **THEN** *buffer*:= *pos1* *pos1*:= *pos2* *pos2*:= *buffer*		Swap order of slot positions if necessary.
Comments: Chooses two random slot positions from the incremental broadcast candidate range used in a number of test cases.						

7.4.3.2.2 Test case descriptions

Test Case Name:	**Physical_SysParams**
Purpose:	**To demonstrate that a station operates correctly at the limits of the physical layer system parameters.**
Special test instructions:	This test case tests requirements in EN 302 842-1 [4]. The test description is located in the present document and not in EN 302 842-1 [4] as the present document includes all the necessary additional requirements and information for the completion of protocol tests, of which this is one. This test case is set up to last up to 10 minutes. Whilst the loop is executing, the timing of the test signals relative to UTC time shall be adjusted using a waveform analyser to inspect the timing between a) the end of the final data bit of the sync burst (t_sync_burst) received in step 19 and the start of the first data bit of the unicast burst (t_unicast1) in step 20, and b) the end of the final bit of the unicast burst (t_unicast2) in step 20 and the start of the first bit of the random access burst in the next slot (t_random). Two tests shall be carried out using this test case: The timing shall be adjusted such that t_unicast1 - t_sync_burst equals 3 083,3 microseconds +/- 1 microseconds. Successful demonstration of the test case step 22 ensures compliance with the present document and (ICAO VDL SARPS [1] 6.9.5.4.3) The timing shall be adjusted such that t_random - t_unicast2 equals 2 099,3 microseconds +/- 1 microseconds. Successful demonstration of the test case step 22 ensures compliance with the present document and (ICAO VDL SARPS [1] 6.9.5.4.1) Note that the tests are carried out using the start and end of the burst data blocks as reference points. This is because it is otherwise impossible to determine the exact end points of the receive function and the start of the transmitter power stabilization sequence.

Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS (p:= 1)		Ensure 100 % chance of transmission on access.
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	4	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_n (Q4:= 3; $TV11_{min}$:= 15; $TV11_{max}$:= 16; V11:= 60; V12:= (2/M1) × 60)	Sn	Set up a series of periodic streams of one-slot messages from the station under test. Q4 set to 3. TV11 reservation hold timer is set to hold stream for 15 superframes. V11 set to 60 bursts within M1 slots. V12 set to give dither range of ±1. Length of sync burst is extended by 2 octets greater than normal 1 slot length.
	5	rep 60		n:= 1		Repeat 60 times to record the times of the sync bursts within the first minute.
	6	await	RF	SYNC_BURST_n (pt:= 0; s= add_A)	Sn	
	7	record	RF	sync_time(n):= time at beginning of slot of n^{th} SYNC_BURST_n diff_time:= sync_time(n) - sync_time(1) - (n - 1) slot_diff(n):= diff_time × M1/60	Sn	Record the time of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each ct_slot and the ct_slot of the first burst and transpose to a common time reference. Convert time differences to slot differences.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	8	endrep		n:= n + 1		
	9	rep 10		m:= 1		Repeat test enough times to allow test station timing to be adjusted. 10 minutes is provided.
	10	rep 30		n:= 1		Repeat 30 times per minute.
	11	send	RF	UNI_BURST_a (sdf:= 1; ro:= slot_diff(n + 1) - 1; lg:= 0; pr:= 0; s:= add_B; d:= add_A) in slot beginning at time = sync_time(n) + 60 × m + 60/(M1)	Ua	Send a unicast burst from a simulated station B, in the slot following the sync burst from station under test, reserving a slot (r_slot) 1 slot after the next sync burst for the source to transmit in (r_slot = t_slot + ro + 1).
	12	macro		M_RAND_ACC_SL (slots:= 100)		Queue random access transmissions over 100 slots.
	13	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	First random access transmission should occur before the slot reserved by the unicast reservation.
	14	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s= add_A)	Ra	Define a reference time to measure relative times from during the test.
	15	repx		q:= 1		
	16	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = start_time + q × 60/M1	Ra	Verify that random access transmissions are made by the station under test in the slots preceding the reserved slot.
	17	until		time = sync_time(n + 1) + 60 × m - 60/M1; q:= q + 1		End the loop when the slot immediately preceding the next sync burst from station under test.
	18	await		time = sync_time(n + 1) + 60 × m		
	19	verify	RF	SYNC_BURST_b (s= add_A) in slot beginning at time = sync_time(n + 1) + 60 × m		
	20	send	RF	UNI_BURST_a (sdf:= 1; ro:= 1; lg:= 0; pr:= 0; s:= add_B; d:= add_A) in slot beginning at time = sync_time(n + 1) + 60 × m + 60/M1	Ua	Send a unicast burst from a simulated station B, reserving a slot 2 slots in the future.
	21	repx		q:= 1		
	22	 verify verify	 RF RF	IF q = 2 THEN no transmission present in slot beginning at time = sync_time(n +1) + 60 × m + (q + 1) × 60/M1 ELSE RAND_ACC_DATA_a (s=add_A) in slot beginning at time = sync_time(n +1) + 60 × m + (q + 1) × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the reserved slot.
	23	until		time = start_time + 100 × 60/M1; q:= q + 1		End the loop 100 slots after the first random access transmission was sent.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	24	endrep		n:= n + 2		
	25	endrep		m:= m + 1		
postamble	26	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	27	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:				**Timing_Primary**		
Purpose:				**To demonstrate that when primary timing is available, a transmission from the station complies with primary timing performance.**		
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	do	TIME	ESTABLISH PRIMARY TIME INPUT		Establish source of primary time information.
test body	3	rep 10		*n*:= 1		
	4	await	RF	SYNC_BURST_c (s = *add_A*)	Sc	Wait for an autonomous sync burst.
	5	verify	RF	For SYNC_BURST_c (s = *add_A*) tfom = 0 or 1	Sc	Verify that the time figure of merit of the autonomous sync burst indicates either certified or non-certified primary time.
	6	record	RF	*t*:= time at which first data is transmitted in the slot containing the sync burst, measured from the test equipment's UTC slot start time		
	7	verify	RF	*t* = 2 083,3 ± 1,1 μs		Verify that the time at which data is first transmitted in the slot is compliant with the requirements of primary timing.
	8	endrep		*n*:= *n* + 1		
postamble	9					Bring test equipment into idle state.
Comments: The first bit of data is required to be transmitted within ±0,6 μs from the start of the slot. The primary time source is required to be synchronized to UTC time with a precision of 0,4 μs two sigma, and can thus be expected to be within 0,5 μs on 99 % of occasions. Thus, the worst case timing error at the RF PCO is expected to be within 0,6 + 0,5 = 1,1 μs. Note that the test equipment's UTC time needs to be certified to be accurate by some means.						

Test Case Name:	**Timing_Secondary**					
Purpose:	**To demonstrate that when primary timing is unavailable, a transmission from the station complies with secondary timing performance.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	do	TIME	DISCONNECT PRIMARY TIME INPUT		Disconnect source of primary time.
test body	3	rep 10		*n*:= 1		
	4	send	RF	SYNC_BURST_a (tfom= 0; s = *add_B*; CPR_LAT(0); lon:= CPR_LON(E 10 NM))	Sa	Send a sync burst from a simulated station B declaring certified primary time. The start of the burst shall be delayed from the slot start time to simulate the delay caused by the time-of-flight from station B.
	5	await	RF	SYNC_BURST_c (s = *add_A*)	Sc	Wait for an autonomous sync burst.
	6	record	RF	*tfom_A* = tfom contained in SYNC_BURST_c (s = *add_A*)	Sc	Time figure of merit of the autonomous sync burst.
	7	record	RF	*t*:= time at which first data is transmitted in the slot containing the sync burst, measured from the test equipment's UTC slot start time		
	8	verify	RF	**IF** *tfom_A* = 0 or 1 **THEN** *t* = 2 083,3 ± 1,1 μs **ELSE** *tfom_A* = 2 **AND** *t* = 2 083,3 ± 20 μs		Verify that the time at which data is first transmitted in the slot is compliant with the requirements of either primary or secondary timing.
	9	endrep		*n*:= *n* + 1		
postamble	10	do	TIME	ESTABLISH PRIMARY TIME INPUT		Re-establish source of primary time information.
Comments: The first bit of data is required to be transmitted within ±0,6 μs from the start of the slot. The secondary time source is required to be synchronized to UTC time with a precision of 15 μs two sigma, and can thus be expected to be within 19,3 μs on 99 % of occasions. Thus, the worst case timing error at the RF PCO is expected to be within 0,6 + 19,3 ≈ 20 μs.						

Test Case Name:	**Timing_Secondary_Recover**					
Purpose:	**To demonstrate that when primary timing becomes available to a station which is transmitting on secondary time, it reverts to using primary time.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
test body	2	rep 10		*n*:= 1		
	3	do	TIME	DISCONNECT PRIMARY TIME INPUT		Disconnect source of primary time.
	4	send	RF	SYNC_BURST_a (tfom= 0; s = *add_B*; CPR_LAT(0); lon:= CPR_LON(E 10 NM))	Sa	Send a sync burst from a simulated station B declaring certified primary time. The start of the burst shall be delayed from the slot start time to simulate the delay caused by the time-of-flight from station B.
	5	await	RF	SYNC_BURST_c (s = *add_A*)	Sc	Wait for an autonomous sync burst.
	6	verify	RF	For SYNC_BURST_c (s = *add_A*) tfom = 0, 1 or 2	Sc	Verify that the time figure of merit of the autonomous sync burst indicates primary or secondary time.
	7	do	TIME	ESTABLISH PRIMARY TIME INPUT		Establish source of primary time.
	8	verify	RF	For SYNC_BURST_c (s = *add_A*) tfom = 0 or 1	Sc	Verify that the time figure of merit of the autonomous sync burst indicates primary time.
	9	record	RF	*t*:= time at which first data is transmitted in the slot containing the sync burst, measured from the test equipment's UTC slot start time		
	10	verify	RF	*t* = 2 083,3 ± 1,1 μs		Verify that the time at which data is first transmitted in the slot is compliant with the requirements of primary timing.
	11	endrep		*n*:= *n* + 1		
postamble	12					Bring test equipment into idle state.
Comments:						

Test Case Name:	CRC_Norm					
Purpose:	**To demonstrate that a station transmitting a burst will insert a valid CRC.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	3	rep 10		n:= 1		
	5	queue	VSS	DATA_a(*m*)	Da(*m*)	Send a packet of data (labelled DATA_a) to the station under test for subsequent transmission by the random access protocol. Identify packets with repeating 10101010 bit sequence over *m* bits
	6	await	RF	RAND_ACC_DATA_a (s = *add_A*)	Ra	Await random access transmission containing DATA a(*m*).
	7	verify	RF	c in RAND_ACC_DATA_a (s = *add_A*) has the correct value according to the algorithm specified in ISO/IEC 13239 [3]	Ra	Verify that the CRC code in the transmitted burst corresponds to the correct value according to the algorithm specified in ISO/IEC 13239 [3].
	8	endrep		n:= n + 1		
postamble	9	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	10	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	CRC_Rej					
Purpose:	**To demonstrate that a station receiving a burst with an invalid CRC will reject the burst.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; c:= invalid; s= add_B)	Sb	Send a sync burst from a simulated station B reserving a slot with an invalid CRC.
	5	macro		M_RAND_ACC_SU (*sf*:= 2)		Queue random access transmissions over 2 superframes.
	6	await	RF	RAND_ACC_DATA_a (s = *add_A*)	Ra	Await random access transmission containing DATA a(*m*).
	7	record	RF	*start_time*:= time at beginning of slot containing RAND_ACC_DATA_a (s = *add_A*)	Ra	Define a reference time to measure relative times from during the test.
	8	rep 2xM1		*n*:= 1		
	9	verify	RF	RAND_ACC_DATA_a (s = *add_A*) in slot beginning at *time* = *start_time* + *n* x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots over 2 superframes.
	10	endrep		*n*:= *n* + 1		
postamble	11	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	12	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Version_NonZero**					
Purpose:	**To demonstrate that a station receiving a burst containing a non-zero version number will ignore the burst and inform the VSS user.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_f(*2*) (pt:= 3; po:= 0; s = *add_B*)	Sf(*2*)	Send a sync burst from a simulated station B reserving a block of slots but with a non-zero version number (ver set to 001binary). Information field filled with zeros extending reservation over 2 slots (burst length = *2*).
	5	macro		M_RAND_ACC_SU (*sf*:= 2)		Queue random access transmissions over 2 superframes.
	6	await	RF	RAND_ACC_DATA_a (s = *add_A*)	Ra	Await random access transmission containing DATA_a(*m*).
	7	record	RF	*start_time*:= time at beginning of slot containing RAND_ACC_DATA_a (s = *add_A*)	Ra	Define a reference time to measure relative times from during the test.
	8	rep 2xM1		*n*:= 1		
	9	verify	RF	RAND_ACC_DATA_a (s = *add_A*) in slot beginning at *time* = *start_time* + *n* x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots over 2 superframes.
	10	endrep		*n*:= *n* + 1		
	11	verify	VSS	Non-zero version number error message		Verify VSS user informed of receipt of reservation with non-zero version number.
postamble	12	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	13	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Queue_Replace					
Purpose:	**To demonstrate that a burst submitted to the VSS layer with Q3 set to TRUE will replace any queued data of the same type.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (p:= 1; Q3:= TRUE)		Ensure 100 % chance of transmission on access.
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	4	send	RF	SYNC_BURST_d(16) (pt:= 3; po:= 0; s:= *add_B*)	Sd(16)	Send a sync burst from a simulated station B extending over 16 slots.
	5	record	RF	*sync_time*:= time at beginning of first slot occupied by SYNC_BURST_d(16) (s = *add_A*)	Sd(16)	Record the time at the start of the sync burst.
	6	await		*time*:= *sync_time* + 60		The reservation from the sync burst of station B prevents the station under test from transmitting for the next 16 slots.
	7	send	VSS	REQUEST TO TRANSMIT SYNC_BURST_l (b/g:= 0) BY RANDOM ACCESS	Sl	Queue a request for transmission by random access of a sync burst with b/g set to 0.
	8	send	VSS	REQUEST TO TRANSMIT SYNC_BURST_l (b/g:= 1) BY RANDOM ACCESS	Sl	Queue a request for transmission by random access of a sync burst with b/g set to 1.
	9	await		*time*:= *sync_time* + 77		Wait until the channel is free of reservations.
	10	verify	RF	SYNC_BURST_l (s = *add_A; b/g = 1*) transmitted **AND** SYNC_BURST_l (s = *add_A; b/g = 0*) not transmitted after *time*:= *sync_time* + 77	Sl	Verify that only the second sync burst is transmitted by the station under test.
postamble	11	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	12	send	VSS	SET PARAMETERS (p:= 64/256; Q3:= FALSE)		Reset to default value.
Comments:						

Test Case Name:	**Queue_Norm**					
Purpose:	**To demonstrate that a burst submitted to the VSS layer with Q3 set to FALSE will not replace any queued data of the same type.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (p:= 1)		Ensure 100 % chance of transmission on access. Q3 set to FALSE by default.
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	4	send	RF	SYNC_BURST_d(16) (pt:= 3; po:= 0; s:= *add_B*)	Sd(16)	Send a sync burst from a simulated station B extending over 16 slots.
	5	record	RF	*sync_time*:= time at beginning of first slot occupied by SYNC_BURST_d(16) (s = *add_A*)	Sd(16)	Record the time at the start of the sync burst.
	6	await		*time*:= *sync_time* + 60		The reservation from the sync burst of station B prevents the station under test from transmitting for the next 16 slots.
	7	send	VSS	REQUEST TO TRANSMIT SYNC_BURST_l (b/g:= 0) BY RANDOM ACCESS	Sl	Queue a request for transmission by random access of a sync burst with b/g set to 0.
	8	send	VSS	REQUEST TO TRANSMIT SYNC_BURST_l (b/g:= 1) BY RANDOM ACCESS	Sl	Queue a request for transmission by random access of a sync burst with b/g set to 1.
	9	await		*time*:= *sync_time* + 60 + 17*60/M1		Wait until the channel is free of reservations.
	10	verify	RF	SYNC_BURST_l (s = *add_A; b/g = 0*) transmitted **AND** SYNC_BURST_l (s = *add_A; b/g = 1*) transmitted after *time*:= *sync_time* + 60 + 17*60/M1	Sl	Verify that both sync bursts are transmitted by the station under test.
postamble	11	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	12	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
Comments:						

Test Case Name:	**MessageID_Invalid_A**					
Purpose:	**To demonstrate that a unicast burst received with an invalid message ID will cause a General Failure burst to be transmitted.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	RF	UNI_BURST_b (sdf:= 0; ro:= 50; lg:= 0; pr:= 0; s:= *add_B*; d:= *add_A*)	Ub	Send a unicast reservation from station B with message ID set to an invalid value.
	4	record	RF	*uni_time*:= time at beginning of first slot occupied by UNI_BURST_b (s = *add_B*)	Ub	Record the time at the start of the unicast burst.
	5	await		*time*:= *uni_time* + 51 x 60/M1		Wait for the slot reserved by the unicast reservation.
	6	verify	RF	GEN_RESP_a (ok= 0; r-mi= 1010101binary; err= 00 hex; bd = 0; s = *add_A*; d:= *add_B*) in slot beginning at *time*:= *uni_time* + 51 x 60/M1	GRa	Verify that a General Failure burst is sent in the slot reserved by the unicast reservation.
postamble	7	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**MessageID_Invalid_B**					
Purpose:	**To demonstrate that a burst with an invalid message ID not making a reservation for reply, causes no response to be made.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	RF	INCREM_BURST_c (io:= 10; s:= *add_B*)	Ic	Send an incremental broadcast reservation from station B with message ID set to an invalid value.
	4	record	RF	*increm_time*:= time at beginning of first slot occupied by INCREM_BURST_c (s = *add_B*)	Ic	Record the time at the start of the incremental burst.
	5	rep M1		*n*:= 1		Wait for the slot reserved by the incremental reservation.
	6	verify	RF	No response from the station under test in slot beginning at *time*:= *increm_time* + *n* x 60/M1		Verify that no response is made by the station under test in the following superframe.
	7	endrep		*n*:= *n* + 1		
postamble	8	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Reservation_Unrecognized**					
Purpose:	**To demonstrate that an unrecognized reservation type will cause the packet to be rejected and an error logged.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	BURST_UNREC_a (s = *add_B*)	Ba	Send a burst from a simulated station B with extended reservation ID (erid) field set to 00111binary, incremental offset (io) field set to 255, and reservation ID (rid) set to 0. The value of the extended reservation ID is currently reserved for future allocation and does not denote a recognized reservation type. The burst also resembles an incremental broadcast reservation with io = 255, reserving a slot 13,6 s later but with the erid field incorrectly set.
	5	macro		M_RAND_ACC_SU (*sf*:= 2)		Queue random access transmissions over 2 superframes.
	6	await	RF	RAND_ACC_DATA_a (s = *add_A*)	Ra	Await random access transmission containing DATA_a(*m*). The first random access transmission shall be within 13 s of the unrecognized reservation burst for the test to be valid.
	7	record	RF	*start_time*:= time at beginning of slot containing RAND_ACC_DATA_a (s = *add_A*)	Ra	Define a reference time to measure relative times from during the test.
	8	rep 2 x M1		*n*:= 1		
	9	verify	RF	RAND_ACC_DATA_a (s:= *add_A*) in slot beginning at *time* = *start_time* + *n* x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots over 2 superframes.
	10	endrep		*n*:= *n* + 1		
	11	verify	VSS	Unrecognized reservation type error message		Verify VSS user informed of receipt of reservation with an unrecognized extended reservation id field.
postamble	12	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	13	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:				**Reservation_Invalid**		
Purpose:				**To demonstrate that reception of a known reservation type with an invalid subfield causes the appropriate slots to be reserved, but not to transmit a response, nor pass the burst to a VSS user.**		
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.
test body	4	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 200 NM)) (position of mobile B is > Q2a away from station under test)	Sa	Send a sync burst from a simulated station B with position data showing that it is > Q2a away from the station under test.
	5	send	RF	UNI_BURST_a (sdf:= 0; ro:= 100; lg:= 0; pr:= 0; s:= add_B; d:= all zeros)	Ua	Send a unicast burst from station B, with sdf = 0, reserving a slot for the destination to transmit. The destination address is set to all zeros, which is invalid.
	6	record	RF	uni_time:= time at beginning of slot containing UNI_BURST_a	Ua	
	7	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	8	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	9	verify	RF	No transmission by station under test in slot beginning at time = uni_time + 101		Verify that no transmission is made by the station under test in the slot reserved by the unicast reservation.
postamble	10	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	11	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:				**Reservation_Recognition**		
Purpose:				**To demonstrate that a reservation will be recognized prior to the end of the slot following the transmission in which it was carried.**		
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	UNI_BURST_a (sdf:= 1; ro:= 2 000; lg:= 0; pr:= 0; s:= *add_B*; d:= *add_A*)	Ua	Send a unicast burst from a simulated station B, reserving a slot (r_slot) 2 001 slots after the transmission slot (t_slot) for the source to transmit in (r_slot = t_slot + ro + 1).
	5	record	RF	*reserve_time*:= time at beginning of slot containing UNI_BURST_a	Ua	Record the time of the slot containing the unicast reservation (*reserve_time* is the time at the beginning of t_slot).
	6	macro		M_RAND_ACC_SU (*sf*:= 1)		Queue random access transmissions over 1 superframe.
	7	await	RF	RAND_ACC_DATA_a (s = *add_A*)	Ra	First random access transmission shall occur before the slot reserved by the unicast reservation.
	8	record	RF	*start_time*:= time at beginning of slot containing RAND_ACC_DATA_a (s = *add_A*)	Ra	Define a reference time to measure relative times from during the test.
	9	repx		*n*:= 1		
	10	verify	RF	RAND_ACC_DATA_a (s = *add_A*)	Ra	Verify that random access transmissions are made by the station

Test Case Name:	Reservation_Recognition					
Purpose:	**To demonstrate that a reservation will be recognized prior to the end of the slot following the transmission in which it was carried.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
				in slot beginning at *time* = *start_time* + *n* x 60/M1		under test in the slots preceding the reserved slot.
	11	until		*time* = *reserve_time* + 2 000 x 60/M1 (in previous step); *n*:= *n* + 1		End the loop when the slot immediately preceding the reserved slot is reached in the loop and checked for data.
	12	await		*time* = *reserve_time* + 2 001 x 60/M1		
	13	send	RF	UNI_BURST_a (sdf:= 1; ro:= 1; lg:= 0; pr:= 0; s:= *add_B*; d:= *add_A*) in slot beginning at *time* = *reserve_time* + 2 001 x 60/M1	Ua	Send a unicast burst from a simulated station B, reserving a slot 2 slots in the future.
	14	repx		*n*:= 1		
	15	 verify verify	 RF RF	**IF** *n* = 2 **THEN** no transmission present in slot beginning at *time* = *reserve_slot* + (*n* + 2 001) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = *add_A*) in slot beginning at *time* = *reserve_slot* + (*n* + 2 001) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the reserved slot.
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	16	until		*time* = *start_time* + 60; *n*:= *n* + 1		End the loop 1 minute after the first random access transmission was sent. Verification therefore takes place over 1 superframe + 1 slot.
postamble	17	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	SlotSel_Level0_A					
Purpose:	**To demonstrate that a station will select a slot at level 0 when no slots are reserved.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b (Q4:= 11; $TV11_{min}$:= 1; $TV11_{max}$:= 1; V11:= 10; V12:= (10/M1) x V11)		Set up a series of periodic streams of one-slot messages from the station under test. Q4 set to 11; equals number of slots in dither range available for selection. TV11 reservation hold timer set to force dither in next frame. V11 set to 10 bursts within M1 slots.

Test Case Name:	SlotSel_Level0_A					
Purpose:	To demonstrate that a station will select a slot at level 0 when no slots are reserved.					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
						V12 set to give dither range of ±5.
	4	rep 111		n:= 1		Repeat test 111 times to generate statistical sample.
	5	await	RF	SYNC_BURST_b (pt:= 0; s = add_A)	Sb	
	6	record	RF	sync_time(n):= time at beginning of slot of n^{th} SYNC_BURST_b diff_time:= sync_time(n) - sync_time(1) - (n - 1) x 6 slot_diff(n):= diff_time x M1/60	Sb	Record the time of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each ct_slot and the ct_slot of the first burst and transpose to a common time frame. Convert time differences to slot differences.
	7	endrep		n:= n + 1		
	8	verify		MAX(slot_diff(n)) - MIN(slot_diff(n)) ≤ V12 x M1/V11		Verify distribution of slots is over candidate slot range.
	9	record		num_slot_diff(m):= 0 for all m		Initialize the number of slots in each candidate slot position to zero.
	10	rep 111		n:= 2		
	11	record		num_slot_diff(slot_diff(n)):= num_slot_diff(slot_diff(n)) + 1		Record the frequency of occurrence of slots in each candidate slot position.
	12	endrep		n:= n + 1		
	13	rep m		m:= MIN(slot_diff(n)); chi_squared:= 0		Set initial value of m to the minimum value of slot_diff.
	14	record		chi_squared:= chi_squared + $(num\_slot\_diff(m) - 10)^2/10$		The distribution is tested for uniformity by calculating the value of chi_squared.
	15	until		m:= MAX(slot_diff(n))		
	16	verify		chi_squared < 21,2		Value of chi_squared shall be less than 21,2 for confidence that the distribution is uniform (10 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
postamble	17	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	18	send	VSS	SET PARAMETERS (Q4:= 3; $TV11_{min}$:= 4; $TV11_{max}$:= 8; V11:= 1; V12:= 0.1)		Reset to default values.
	19	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	SlotSel_Level0_B					
Purpose:	To demonstrate that a station will select a slot at level 0, excluding those not meeting the criteria of any other level.					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value

Test Case Name:	SlotSel_Level0_B					
Purpose:	**To demonstrate that a station will select a slot at level 0, excluding those not meeting the criteria of any other level.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
						of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		random_position:= 64 + 4 x RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station B is < Q2a, b, c, d away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B < Q2a, b, c, d away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	record		reserve_slot:= 4 x IO(n - 1) + random_position		Slot position to reserve within the next-but-one incremental broadcast candidate range.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	13	send	RF	INCREM_BURST_a (io:= (reserve_slot - 16)/4; s:= add_B) in slot beginning at time = current_inc_time + 16 x 60/M1	Ia	Send a broadcast burst from station B < Q2a, b, c, d away from A. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	14	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	15	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	16	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	17	endrep		n:= n + 1		
	18	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station B.
	19	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	20	 record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	21	endrep		m:= m + 4		
	22	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	23	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	24	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level0_C**					
Purpose:	**To demonstrate that a station will select a slot at level 0 in preference to those slots available at level 1.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals one less than the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		random_position:= 64 + 4 x RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2a away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 170 NM)) (position of station D is such that a transmission from B to D will be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected.
	13	record		reserve_slot:= 4 x IO(n - 1) + random_position		Slot position to reserve within the next-but-one incremental broadcast candidate range.
	14	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2a away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	15	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	16	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	17	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	18	endrep		n:= n + 1		
	19	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station B.
	20	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	21	 record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	22	endrep		m:= m + 4		
	23	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	24	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	25	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level0_D					
Purpose:	**To demonstrate that a station will select a slot at level 0 in preference to those slots available at level 2.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals one less than the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		random_position:= 64 + 4 x RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2b away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2b away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	record		reserve_slot:= 4 x IO(n - 1) + random_position		Slot position to reserve within the next-but-one incremental broadcast candidate range.
	13	send	RF	INCREM_BURST_a (io:= (reserve_slot - 16)/4; s:= add_B + n x 1 binary) in slot beginning at time = current_inc_time + 16 x 60/M1	Ia	Send a broadcast burst from station B > Q2b away from A. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	14	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	15	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	16	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	17	endrep		n:= n + 1		
	18	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station B.
	19	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	20	record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + (no_IO(m) - 10)^2/10		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	21	endrep		m:= m + 4		
	22	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	23	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	24	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level0_E					
Purpose:	**To demonstrate that a station will select a slot at level 0 in preference to those slots available at level 3.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals one less than the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		random_position:= 64 + 4 x RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 130 NM)) (position of station B is > Q2c away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2c away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station D is such that a transmission from B to D will be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected.
	13	record		reserve_slot:= 4 x IO(n - 1) + random_position		Slot position to reserve within the next-but-one incremental broadcast candidate range.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2c away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	15	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	16	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	17	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	18	endrep		n:= n + 1		
	19	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station B.
	20	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	21	record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	22	endrep		m:= m + 4		
	23	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	24	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	25	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level0_F**					
Purpose:	**To demonstrate that a station will select a slot at level 0 in preference to those slots available at level 4.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals one less than the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		random_position:= 64 + 4 x RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 310 NM)) (position of station B is > Q2d away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2d away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station D is such that a transmission from B to D will not be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will not be CCI protected.
	13	record		reserve_slot:= 4 x IO(n - 1) + random_position		Slot position to reserve within the next-but-one incremental broadcast candidate range.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2d away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will not be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	15	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	16	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	17	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	18	endrep		n:= n + 1		
	19	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station B.
	20	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	21	 record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	22	endrep		m:= m + 4		
	23	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	24	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	25	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level1_A					
Purpose:	**To demonstrate that a station will select a slot at level 1 when the appropriate criteria are satisfied.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		random_position:= 64 + 4 x RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 60		n:= 1		Repeat 60 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2a away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 170 NM)) (position of station D is such that a transmission from B to D will be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected.
	13	record		reserve_slot:= 4 x IO(n - 1) + random_position		Slot position to reserve within the next-but-one incremental broadcast candidate range.
	14	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2a away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	15	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	16	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	17	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	18	endrep		n:= n + 1		
	19	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	20	record		chi_squared:= chi_squared + $(\text{no\_IO}(m) - 10)^2/10$		The distribution is tested for uniformity by calculating the value of chi_squared.
	21	endrep		m:= m + 4		
	22	verify		chi_squared < 13,4		Value of chi_squared shall be less than 13,4 for confidence that the distribution is uniform (5 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	23	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	24	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level1_B**					
Purpose:	**To demonstrate that a station will select a slot at level 1, excluding those slots not meeting the criteria of level 1 or any lower priority level.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2a away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station E is < Q2a, b, c, d away from station under test) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E < Q2a, b, c, d away from the station under test, reporting E's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 130 NM)) (position of station D is such that a transmission from B to D will be CCI protected and that a transmission from E to D will be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected and that a transmission from E to D will be CCI protected.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_1 - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2a away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E < Q2a,b,c,d away from A, reserving a slot for transmission from station E to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	 record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level1_C**					
Purpose:	**To demonstrate that a station will select a slot at level 1 in preference to those available at level 2.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals one less than the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2a away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 170 NM)) (position of station E is > Q2b away from station under test) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E > Q2b away from the station under test, reporting E's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 180 NM)) (position of station D is such that a transmission from B to D will be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected.
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_1 - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2a away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	INCREM_BURST_a (io:= (reserve_slot_2 - 20)/4; s:= add_E + n x 1 binary) in slot beginning at time = current_inc_time + 20 x 60/M1	Ia	Send a broadcast burst from station E > Q2b away from A. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level1_D**					
Purpose:	**To demonstrate that a station will select a slot at level 1 in preference to those available at level 3.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2a away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 130 NM)) (position of station E is > Q2c away from station under test) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E > Q2c away from the station under test, reporting E's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station D is such that a transmission from B to D will be CCI protected and that a transmission from E to D will be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected and that a transmission from E to D will be CCI protected.
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_1 - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2a away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E > Q2c away from A, reserving a slot for transmission from station E to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level1_E**					
Purpose:	**To demonstrate that a station will select a slot at level 1, in preference to those available at level 4.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2a away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 310 NM)) (position of station E is > Q2d away from station under test) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E > Q2d away from the station under test, reporting E's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station D is such that a transmission from B to D will be CCI protected and that a transmission from E to D will not be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected and that a transmission from E to D will not be CCI protected.
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_1 - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2a away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E > Q2d away from A, reserving a slot for transmission from station E to station D. The distance from the station under test (station A) to station D is < (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will not be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	23	 record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + (no_IO(m) - 10)^2/10		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level1_F**					
Purpose:	**To demonstrate that a station will select slots at level 1 from a more distant station in preference to a closer station.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 170 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2a away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station E is > Q2a away from station under test but closer to the station under test than station B) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E > Q2a away from the station under test, reporting E's position. Station E is closer to the station under test than station B. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 180 NM)) (position of station D is such that a transmission from B to D will be CCI protected and that a transmission from E to D will be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected and that a transmission from E to D will be CCI protected.
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_1 - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2a away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E > Q2a away from A, reserving a slot for transmission from station E to station D. Station E is closer to the station under test than station B. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	 record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level2_A**					
Purpose:	**To demonstrate that a station will select a slot at level 2 when the appropriate criteria are satisfied.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		random_position:= 64 + 4 x RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 60		n:= 1		Repeat 60 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2b away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2b away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	record		reserve_slot:= 4 x IO(n - 1) + random_position		Slot position to reserve within the next-but-one incremental broadcast candidate range.
	13	send	RF	INCREM_BURST_a (io:= (reserve_slot - 16)/4; s:= add_B + n x 1 binary) in slot beginning at time = current_inc_time + 16 x 60/M1	Ia	Send a broadcast burst from station B > Q2b away from A. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	14	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	15	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	16	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	17	endrep		n:= n + 1		
	18	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	19	record		chi_squared:= chi_squared + (no_IO(m) - 10)^2/10		The distribution is tested for uniformity by calculating the value of chi_squared.
	20	endrep		m:= m + 4		
	21	verify		chi_squared < 13,4		Value of chi_squared shall be less than 13,4 for confidence that the distribution is uniform (5 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	22	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	23	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level2_B					
Purpose:	**To demonstrate that a station will select a slot at level 2, excluding those slots not meeting the criteria of level 2 or any lower priority level.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2b away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2b away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station E is < Q2a, b, c, d away from station under test) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E < Q2a, b, c, d away from the station under test, reporting E's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	14	send	RF	INCREM_BURST_a (io:= (reserve_slot_1 - 16)/4; s:= add_B + n x 1 binary) in slot beginning at time = current_inc_time + 16 x 60/M1	Ia	Send a broadcast burst from station B > Q2b away from A. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	15	send	RF	INCREM_BURST_a (io:= (reserve_slot_2 - 20)/4; s:= add_E + n x 1 binary) in slot beginning at time = current_inc_time + 20 x 60/M1	Ia	Send a broadcast burst from station E < Q2a,b,c,d away from A. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	17	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	18	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	19	endrep		n:= n + 1		
	20	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	21	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	22	record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	23	endrep		m:= m + 4		
	24	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	25	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	26	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level2_C					
Purpose:	**To demonstrate that a station will select a slot at level 2 in preference to those available at level 3.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2b away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 130 NM)) (position of station E is > Q2c away from station under test) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E > Q2c away from the station under test, reporting E's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station D is such that a transmission from E to D will be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from E to D will be CCI protected.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	INCREM_BURST_a (io:= (reserve_slot_1 - 16)/4; s:= add_B + n x 1 binary) in slot beginning at time = current_inc_time + 16 x 60/M1	Ia	Send a broadcast burst from station B > Q2b away from A. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E > Q2c away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	 record		**IF** m $\neq$ random_position **THEN** chi_squared:= chi_squared + (no_IO(m) - 10)^2/10		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level2_D					
Purpose:	**To demonstrate that a station will select a slot at level 2 in preference to those available at level 4.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2b away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 310 NM)) (position of station E is > Q2d away from station under test) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E > Q2d away from the station under test, reporting E's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station D is such that a transmission from E to D will not be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from E to D will not be CCI protected.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	INCREM_BURST_a (io:= (reserve_slot_1 - 16)/4; s:= add_B + n x 1 binary) in slot beginning at time = current_inc_time + 16 x 60/M1	Ia	Send a broadcast burst from station B > Q2b away from A. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E > Q2c away from A, reserving a slot for transmission from station E to station D. The distance from the station under test (station A) to station D is < (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will not be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level2_E**					
Purpose:	**To demonstrate that a station will select slots at level 2 from a more distant station in preference to a closer station.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 170 NM)) (position of station B is > Q2b away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2b away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station E is > Q2b away from station under test, but closer to the station under test than station B) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E > Q2b away from the station under test, reporting E's position. Station E is closer to the station under test than station B. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	14	send	RF	INCREM_BURST_a (io:= (reserve_slot_1 - 16)/4; s:= add_B + n x 1 binary) in slot beginning at time = current_inc_time + 16 x 60/M1	Ia	Send a broadcast burst from station B > Q2b away from A. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	15	send	RF	INCREM_BURST_a (io:= (reserve_slot_2 - 20)/4; s:= add_E + n x 1 binary) in slot beginning at time = current_inc_time + 20 x 60/M1	Ia	Send a broadcast burst from station E > Q2b away from A. Station E is closer to the station under test than station B. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	17	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	18	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	19	endrep		n:= n + 1		
	20	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	21	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	22	 record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(\text{no\_IO}(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	23	endrep		m:= m + 4		
	24	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	25	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	26	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level3_A					
Purpose:	**To demonstrate that a station will select a slot at level 3 when the appropriate criteria are satisfied.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		random_position:= 64 + 4 x RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 60		n:= 1		Repeat 60 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 130 NM)) (position of station B is > Q2c away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2c away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	record		reserve_slot:= 4 x IO(n - 1) + random_position		Slot position to reserve within the next-but-one incremental broadcast candidate range.
	13	send	RF	INCREM_BURST_a (io:= (reserve_slot - 16)/4; s:= add_B + n x 1 binary) in slot beginning at time = current_inc_time + 16 x 60/M1	Ia	Send a broadcast burst from station B > Q2c away from A. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	14	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	15	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	16	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	17	endrep		n:= n + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	18	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	19	record		chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		The distribution is tested for uniformity by calculating the value of chi_squared.
	20	endrep		m:= m + 4		
	21	verify		chi_squared < 13,4		Value of chi_squared shall be less than 13,4 for confidence that the distribution is uniform (5 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	22	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	23	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level3_B					
Purpose:	**To demonstrate that a station will select a slot at level 3, excluding those slots not meeting the criteria of level 3 or any lower priority level.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2c away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station E is < Q2a, b, c, d away from station under test) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E < Q2a, b, c, d away from the station under test, reporting E's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 130 NM)) (position of station D is such that a transmission from B to D will be CCI protected and that a transmission from E to D will be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected and that a transmission from E to D will be CCI protected.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_1 - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2c away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E < Q2a, b, c, d away from A, reserving a slot for transmission from station E to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	 record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level3_C**					
Purpose:	**To demonstrate that a station will select a slot at level 3 in preference to those available at level 4.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 130 NM)) (position of station B is > Q2c away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2c away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 310 NM)) (position of station E is > Q2d away from station under test) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E > Q2d away from the station under test, reporting E's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station D is such that a transmission from B to D will be CCI protected and that a transmission from E to D will not be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected and that a transmission from E to D will not be CCI protected.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_1 - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2c away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E > Q2d away from A, reserving a slot for transmission from station E to station D. The distance from the station under test (station A) to station D is < (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will not be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level3_D					
Purpose:	**To demonstrate that a station will select slots at level 3 from a more distant station in preference to a closer station.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station B is > Q2c away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2c away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 130 NM)) (position of station E is > Q2c away from station under test but closer to the station under test than station B) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E > Q2c away from the station under test, reporting E's position. Station E is closer to the station under test than station B. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 150 NM)) (position of station D is such that a transmission from B to D will be CCI protected and that a transmission from E to D will be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will be CCI protected and that a transmission from E to D will be CCI protected.
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_1 - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2c away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E > Q2c away from A, reserving a slot for transmission from station E to station D. Station E is closer to the station under test than station B. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	 record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level4_A					
Purpose:	**To demonstrate that a station will select a slot at level 4 when the appropriate criteria are satisfied.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		random_position:= 64 + 4 x RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 60		n:= 1		Repeat 60 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 310 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2d away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station D is such that a transmission from B to D will not be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will not be CCI protected.
	13	record		reserve_slot:= 4 x IO(n - 1) + random_position		Slot position to reserve within the next-but-one incremental broadcast candidate range.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2d away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is < (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will not be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	15	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	16	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	17	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	18	endrep		n:= n + 1		
	19	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	20	record		chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		The distribution is tested for uniformity by calculating the value of chi_squared.
	21	endrep		m:= m + 4		
	22	verify		chi_squared < 13,4		Value of chi_squared shall be less than 13,4 for confidence that the distribution is uniform (5 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	23	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	24	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Level4_B					
Purpose:	**To demonstrate that a station will select a slot at level 4, excluding those slots not meeting the criteria of level 4.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 310 NM)) (position of station B is > Q2d away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2d away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station E is < Q2a, b, c, d away from station under test) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E < Q2a, b, c, d away from the station under test, reporting E's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 50 NM)) (position of station D is such that a transmission from B to D will not be CCI protected and that a transmission from E to D will not be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will not be CCI protected and that a transmission from E to D will not be CCI protected.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_1 - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2d away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is < (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will not be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E < Q2a, b, c, d away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is < (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will not be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	record		**IF** m ≠ random_position **THEN** chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_Level4_C**					
Purpose:	**To demonstrate that a station will select a slot at level 4 from a more distant station in preference to a closer station.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 5; V22:= 12)		Q4 set to 5; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		M_ASSIGN_SLOTS (random_position_1, random_position_2)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	9	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	10	rep 50		n:= 1		Repeat 50 times.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 320 NM)) (position of station B is > Q2d away from station under test) in slot beginning at time = current_inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2d away from the station under test, reporting B's position. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E + n x 1 binary; lat:= CPR_LAT(0); lon:= CPR_LON(E 310 NM)) (position of station E is > Q2d away from station under test but closer to the station under test than station B) in slot beginning at time = current_inc_time + 7 x 60/M1	Sa	Send a sync burst from a simulated station E > Q2d away from the station under test, reporting E's position. Station E is closer to the station under test than station B. The simulated station's initial 24-bit address is chosen with a sufficiently low value to allow the address to be incremented by 1 binary in each loop, and also avoiding addresses used by station B.
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station D is such that a transmission from B to D will not be CCI protected and that a transmission from E to D will not be CCI protected) in slot beginning at time = current_inc_time + 10 x 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D will not be CCI protected and that a transmission from E to D will not be CCI protected.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	record		reserve_slot_1:= 4 x IO(n - 1) + random_position_1 reserve_slot_2:= 4 x IO(n - 1) + random_position_2		Slot positions to reserve within the next-but-one incremental broadcast candidate range.
	15	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_1 - 15 - 1; lg:= 0; pr:= 0; d:= add_B + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 15 x 60/M1	Ua	Send a unicast burst from station D to station B > Q2d away from A, reserving a slot for transmission from station B to station D. The distance from the station under test (station A) to station D is < (CCI ratio) times the distance from station B to station D, so that the transmission from B to D will not be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	send	RF	UNI_BURST_a (sdf:= 0; ro:= reserve_slot_2 - 20 - 1; lg:= 0; pr:= 0; d:= add_E + n x 1 binary; s:= add_D) in slot beginning at time = current_inc_time + 20 x 60/M1	Ua	Send a unicast burst from station D to station E > Q2d away from A, reserving a slot for transmission from station E to station D. Station E is closer to the station under test than station B. The distance from the station under test (station A) to station D is < (CCI ratio) times the distance from station E to station D, so that the transmission from E to D will not be CCI protected. The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	17	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	18	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	19	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	20	endrep		n:= n + 1		
	21	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station E.
	22	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	23	record		**IF** m $\neq$ random_position **THEN** chi_squared:= chi_squared + (no_IO(m) - 10)^2/10		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	24	endrep		m:= m + 4		
	25	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	SlotSel_Block_Level0_A					
Purpose:	**To demonstrate that a station will select a block of slots at level 0 when no slots are reserved.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_d(2) (Q4:= 10; $TV11_{min}$:= 1; $TV11_{max}$:= 1; V11:= 10; V12:= (10/M1) x V11)	Sd(2)	Set up a series of periodic streams of two-slot messages from the station under test. Q4 set to 10; equals one less than the number of slots in the dither range available for selection. TV11 reservation hold timer set to force dither in next frame. V11 set to 10 bursts within M1 slots. V12 set to give dither range of ±5.
	4	rep 111		n:= 1		Repeat test 111 times to generate statistical sample.
	5	await	RF	SYNC_BURST_d(2) (pt:= 0; s = add_A)	Sd(2)	
	6	record	RF	sync_time(n):= time at beginning of first slot of n^{th} SYNC_BURST_d(2) diff_time:= sync_time(n) - ((n - 1)/10) x 60 - sync_time(1) slot_diff(n):= diff_time x M1/60	Sd(2)	Record the time of the first slot of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each ct_slot and the ct_slot of the first burst and transpose to a common time frame. Convert time differences to slot differences.
	7	endrep		n:= n + 1		
	8	verify		MAX(slot_diff(n)) - MIN(slot_diff(n)) + 1 ≤ V12 x M1/V11		Verify distribution of blocks of slots is over candidate slot range.
	9	record		num_slot_diff(m):= 0 for all m		Initialize the number of blocks of slots in each candidate slot position to zero.
	10	rep 111		n:= 2		
	11	record		num_slot_diff(slot_diff(n)):= num_slot_diff(slot_diff(n)) + 1		Record the frequency of occurrence of blocks of slots in each candidate slot position.
	12	endrep		n:= n + 1		
	13	rep m		m:= MIN(slot_diff(n)); chi_squared:= 0		Set initial value of m to the minimum value of slot_diff.
	14	record		chi_squared:= chi_squared + $(num\_slot\_diff(m) - 10)^2/10$		The distribution is tested for uniformity by calculating the value of chi_squared.
	15	until		m:= MAX(slot_diff(n))		
	16	verify		chi_squared < 21,2		Value of chi_squared shall be less than 21,2 for confidence that the distribution is uniform (10 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	17	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	18	send	VSS	SET PARAMETERS (Q4:= 3; $TV11_{min}$:= 4; $TV11_{max}$:= 8; V11:= 1; V12:= 0,1)		Reset to default values.
	19	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**SlotSel_Block_Level0_B**					
Purpose:	**To demonstrate that a station will select a block of slots at level 0, excluding those not meeting the criteria of any other level.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_d(2) (Q4:= 6; $TV11_{min}$:= 1; $TV11_{max}$:= 1; V11:= 1; V12:= (6/M1) x V11; INFO:= 246 bits of {0})	Sd(2)	Set up a periodic stream of two-slot messages from the station under test. Q4 set to 6; equals one less than the number of slots in the dither range available for selection. TV11 reservation hold timer set to force dither in next frame. V11 set to 1. V12 set to give dither range of ±3.
	4	await	RF	SYNC_BURST_d(2) (s = add_A)	Sd(2)	
	5	record	RF	reserve_time:= time at the beginning of the first slot of SYNC_BURST_d(2) (s = add_A)	Sd(2)	Define a reference time to measure relative times from during the test. This slot position will be used for the reserved slot after the station under test has dithered away from this slot.
	6	await		time = reserve_time + 60 - 50/M1x 60		Wait for reserve_time plus 1 superframe minus 50 slots.
	7	send	RF	SYNC_BURST_a (pt:= 0; po:= 50; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station B is < Q2a, b, c, d away from the station under test) in slot beginning at time = reserve_time + 60 - 50/M1x 60	Sa	Send a sync burst from a simulated station B < Q2a, b, c, d away from the station under test. This sync burst is outside the dither range of the station under test but is set to dither into the reserved slot (which is within the dither range of the station under test) in the following superframe.
	8	await		time = reserve_time + 120		Wait for reserve_time plus 2 superframes.
	9	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station B is < Q2a, b, c, d away from the station under test) in slot beginning at time = reserve_time + 120	Sa	Send a sync burst from station B < Q2a, b, c, d away from the station under test, which reserves the same slot for the following 4 superframes.
	10	await		time = reserve_time + 150		Wait until after the sync burst from the station under test has occurred in the current superframe.
	11	rep p		p:= 0		Start an outer loop that contains a reservation renewal.
	12	rep 2		n:= 1 + (2 x p) q:= 1 + (3 x p)		Start an inner loop that records the times of the sync bursts made by the station under test. The variables are defined to label each recorded time according to the relative superframe in which it occurred. The definition takes into account superframes in which no time is recorded because an action to renew the reservation by station B has been undertaken instead.
	13	await	RF	SYNC_BURST_d(2) (s = add_A)	Sd(2)	

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	record	RF	sync_time(n):= time at beginning of first slot of n^{th} SYNC_BURST_d(2) (s = add_A) diff_time:= sync_time(n) - (q - 1) x 60 - sync_time(1) ct_slot_diff(n):= diff_time x M1/60	Sd(2)	Record the time of the first slot of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each ct_slot and the ct_slot of the first burst and transpose to a common time frame. Convert time differences to slot differences.
	15	endrep		n:= n + 1 q:= q + 1		The inner loop makes recordings for 2 successive frames before exiting to the outer loop that makes an action in the next successive frame.
	16	await		time = reserve_time + 3 x (p + 1) x 60 + 120		Await the third reserved slot out of the four reserved by the last sync burst from station B.
	17	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station B is < Q2a, b, c, d away from the station under test) in slot beginning at time = reserve_time + 3 x (p + 1) x 60 + 120	Sa	Every third superframe, send a sync burst from station B < Q2a, b, c, d away from the station under test, renewing the reservation for another 4 superframes.
	18	until		p:= 29; p:= p + 1		
	19	verify		MAX(ct_slot_diff(n)) - MIN(ct_slot_diff(n)) + 1 ≤ V12 x M1/V11		Verify distribution of blocks of slots is equal to or less than the candidate slot range.
	20	record		no_ct_slot_diff(m):= 0 for all m		Initialize array of variables to store frequency of occurrence of blocks of slots in each candidate slot position.
	21	rep 60		n:= 1		
	22	record		no_ct_slot_diff(ct_slot_diff(n)):= no_ct_slot_diff(ct_slot_diff(n)) + 1		Record the frequency of occurrence of blocks of slots in each candidate slot position.
	23	endrep		n:= n + 1		
	24	record		m_res_slot:= (reserve_time + 180 -sync_time(1)) x M1/60		Calculate relative slot difference between the reserved slot and the reference slot when transposed onto a common frame
	25	verify		no_ct_slot_diff(m_res_slot) = 0 no_ct_slot_diff(m_res_slot - 1) = 0		Verify that no transmission is made in the slot reserved by station B, or in the slot that is one slot before the slot reserved by station B.
	26	rep m		m:= MIN(ct_slot_diff(n)); chi_squared:= 0		Set value of m to the minimum value of ct_slot_diff

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	27	record record		**IF** m ≠ m_res_slot **OR** m ≠ m_res_slot - 1 **THEN** **{** **IF** m_res_slot = MIN(ct_slot_diff(n)) **OR** m_res_slot = MIN(ct_slot_diff(n)) +6 **THEN** chi_squared:= chi_squared + $(no\_ct\_slot\_diff(m) - (10))^2$ /(10) **ELSE** chi_squared:= chi_squared + $(no\_ct\_slot\_diff(m) - (12))^2$ /(12) **}**		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	28	until		m:= MAX(slot_diff(n))		
	29	verify		**IF** m_res_slot = MIN(ct_slot_diff(n)) **OR** m_res_slot = MIN(ct_slot_diff(n)) +6 **THEN** chi_squared < 13,4 **ELSE** chi_squared < 11,7		If the reserved slot is either the first or the last slot in the dither range, then the value of chi_squared shall be less than 13,4 for confidence that the distribution is uniform (5 degrees of freedom). Otherwise the value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	30	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	31	send	VSS	SET PARAMETERS (Q4:= 3; $TV11_{min}$:= 4; $TV11_{max}$:= 8; V11:= 1; V12:= 0,1)		Reset to default values.
	32	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**SlotSel_Block_MixedLevel**					
Purpose:	**To demonstrate that a station will select a block of slots from slots available at different levels.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_d(2) (Q4:= 6; TV11$_{min}$:= 1; TV11$_{max}$:= 1; V11:= 1; V12:= (6/M1) x V11; INFO:= 246 bits of {0})	Sd(2)	Set up a periodic stream of two-slot messages from the station under test. Q4 set to 6; equals one less than the number of slots in the dither range available for selection. TV11 reservation hold timer set to force dither in next frame. V11 set to 1. V12 set to give dither range of ±3.
	4	await	RF	SYNC_BURST_d(2) (s = add_A)	Sd(2)	
	5	record	RF	reserve_time:= time at the beginning of the first slot of SYNC_BURST_d(2) (s = add_A)	Sd(2)	Define a reference time to measure relative times from during the test. This slot position will be used for the reserved slot after the station under test has dithered away from this slot.
	6	await		time = reserve_time + 60 - 50/M1x 60		Wait for reserve_time plus 1 superframe minus 50 slots.
	7	send	RF	SYNC_BURST_a (pt:= 0; po:= 50; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station B is < Q2a, b, c, d away from the station under test) in slot beginning at time = reserve_time + 60 - 50/M1x 60	Sa	Send a sync burst from a simulated station B < Q2a away from the station under test. This sync burst is outside the dither range of the station under test but is set to dither into the reserved slot (which is within the dither range of the station under test) in the following superframe.
	8	await		time = reserve_time + 120		Wait for reserve_time plus 2 superframes.
	9	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station B is < Q2a away from the station under test) in slot beginning at time = reserve_time + 120	Sa	Send a sync burst from station B < Q2a away from the station under test, which reserves the same slot for the following 4 superframes.
	10	await		time = reserve_time + 150		Wait until after the sync burst from the station under test has occurred in the current superframe.
	11	rep p		p:= 0		Start an outer loop that contains a reservation renewal.
	12	rep 3		n:= 1 + (4 x p)		Start an inner loop that records the times of the sync bursts made by the station under test. The variables are defined to label each recorded time according to the relative superframe in which it occurred. The definition takes into account superframes in which no time is recorded because an action to renew the reservation by station B has been undertaken instead.
	13	await	RF	SYNC_BURST_d(2) (s = add_A)	Sd(2)	

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	record	RF	sync_time(n):= time at beginning of first slot of n^{th} SYNC_BURST_d(2) (s = add_A) diff_time:= sync_time(n) - (n - 1) x 60 - sync_time(1) ct_slot_diff(n):= diff_time x M1/60	Sd(2)	Record the time of the first slot of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each ct_slot and the ct_slot of the first burst and transpose to a common time frame. Convert time differences to slot differences.
	15	endrep		n:= n + 1		The inner loop makes recordings for 3 successive frames before exiting to the outer loop that makes an action in the next successive frame.
	16	await		time = reserve_time + 4 x (p + 1) x 60 + 120		Await the last reserved slot out of the four reserved by the last sync burst from station B.
	17	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station B is < Q2a away from the station under test) in slot beginning at time = reserve_time + 4 x (p + 1) x 60 + 120	Sa	Every fourth superframe, send a sync burst from station B < Q2a away from the station under test, renewing the reservation for another 4 superframes.
	18	until		p:= 23; p:= p + 1		
	19	verify		MAX(ct_slot_diff(n)) - MIN(ct_slot_diff(n)) + 1 ≤ V12 x M1/V11		Verify distribution of blocks of slots is equal to or less than the candidate slot range.
	20	record		no_ct_slot_diff(m):= 0 for all m		Initialize array of variables to store frequency of occurrence of blocks of slots in each candidate slot position.
	21	rep 35		n:= 2		
	22	record		no_ct_slot_diff(ct_slot_diff(n)):= no_ct_slot_diff(ct_slot_diff(n)) + 1		Record the frequency of occurrence of blocks of slots in each candidate slot position.
	23	endrep		n:= n + 1		
	24	rep m		m:= MIN(slot_diff(n)); chi_squared:= 0		Set value of m to the minimum value of slot_diff
	25	record		chi_squared:= chi_squared + (no_ct_slot_diff(m) - $(72/7))^2$ /(72/7)		The distribution is tested for uniformity by calculating the value of chi_squared.
	26	until		m:= MAX(slot_diff(n))		
	27	verify		chi_squared < 15.0		Value of chi_squared shall be less than 15.0 for confidence that the distribution is uniform (6 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	28	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	29	send	VSS	SET PARAMETERS (Q4:= 3; $TV11_{min}$:= 4; $TV11_{max}$:= 8; V11:= 1; V12:= 0,1)		Reset to default values.
	30	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	SlotSel_Reselection					
Purpose:	**To demonstrate that a station after selecting a slot which has been reserved by another station will not select a slot which has been reserved by the same station within the next M1-1 slots.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{min}$:= 1; $TV11_{max}$:= 1; V11:= 2; V12:= (2/M1) x V11)	Sb	Set up two periodic streams of one-slot messages from the station under test. Q4 has default value of 3; equals number of slots in dither range available for selection. TV11 reservation hold timer set to force dither in next frame. V11 set to 2 bursts within M1 slots. V12 set to minimum; equals dither range of ±1.
	4	rep 16		n:= 1		Repeat test 16 times to establish boundaries of candidate slot range for the two streams.
	5	await	RF	SYNC_BURST_b (pt = 0; s = add_A)	Sb	Await periodic stream 1.
	6	record	RF	sync_time1(n):= time at beginning of slot of n^{th} SYNC_BURST_b (s = add_A) diff_time:= sync_time1(n) - (n - 1) x 60 - sync_time1(1) slot_diff1(n):= diff_time x M1/60	Sb	Record the time of the n^{th} sync burst. sync_time1(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each ct_slot and the ct_slot of the first burst and transpose to a common time frame. Convert time differences to slot differences.
	7	await	RF	SYNC_BURST_b (pt = 0; s = add_A)	Sb	Await periodic stream 2.
	8	record	RF	sync_time2(n):= time at beginning of slot of n^{th} SYNC_BURST_b (s = add_A) diff_time:= sync_time2(n) - (n - 1) x 60 - sync_time2(1) slot_diff2(n):= diff_time x M1/60	Sb	Record the time of the n^{th} sync burst. sync_time2(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each ct_slot and the ct_slot of the first burst and transpose to a common time frame. Convert time differences to slot differences.
	9	endrep		n:= n + 1		
	10	verify		MAX(slot_diff1(n)) - MIN(slot_diff1(n)) ≤ V12 x M1/V11		Verify distribution of slots is over candidate range for stream 1.
	11	verify		MAX(slot_diff2(n)) - MIN(slot_diff2(n)) ≤ V12 x M1/V11		Verify distribution of slots is over candidate range for stream 2.
	12	record		reserve_time1:= sync_time1(1) + (18 + (MIN(slot_diff1(n))/M1)) x 60 reserve_time2:= sync_time2(1) + (18 + (MIN(slot_diff2(n))/M1)) x 60		Select the first slot in the candidate range to make a reservation.
	13	await		time = reserve_time1 - 50 x 60/M1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	send	RF	SYNC_BURST_d(3) (pt:= 1; po:= 50; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2b away from the station under test) in slot beginning at time = reserve_time1 - 50 x 60/M1	Sd(3)	Send a sync burst from a simulated station B > Q2b away from the station under test. This sync burst is outside the candidate range of stream 1 but is set to dither into the first slot of the candidate range of this stream in the next but one superframe. The burst reserves 3 slots and will thus extend over the whole of the candidate range when it dithers.
	15	await		time = reserve_time2 - 50 x 60/M1		
	16	send	RF	SYNC_BURST_d(3) (pt:= 1; po:= 50; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2b away from the station under test) in slot beginning at time = reserve_time2 - 50 x 60/M1	Sd(3)	Send a sync burst from a simulated station B > Q2b away from the station under test. This sync burst is outside the candidate range of stream 2 but is set to dither into the first slot of the candidate range of this stream in the next but one superframe. The burst reserves 3 slots and will thus extend over the whole of the candidate range when it dithers.
	17	await		time = reserve_time1 + 120		Wait for the beginning of the reservation across the candidate range of stream 1.
	18	verify	RF	SYNC_BURST_b (s = add_A) transmitted before time = reserve_time1 + 3 x 60/M1 + 120	Sb	Verify that a sync burst is transmitted by the station under test within the candidate range of stream 1, even though it conflicts with the reservation made by station B.
	19	await		time = reserve_time2 + 120		Wait for the beginning of the reservation across the candidate range of stream 2.
	20	verify	RF	no SYNC_BURST_b (s = add_A) transmitted before time = reserve_time2 + 3 x 60/M1 + 120	Sb	Verify that no sync burst is transmitted by the station under test in the candidate range of stream 2, and therefore within M1 slots of the last transmission made in a slot reserved by station B.
	21	verify	VSS	no slot available for selection		Verify that the VSS user is informed that no slot was available for selection.
postamble	22	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	23	send	VSS	SET PARAMETERS (Q4:= 3; $TV11_{min}$:= 4; $TV11_{max}$:= 8; V11:= 1; V12:= 0,1)		Reset to default values.
	24	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**SlotSel_Unsuccessful**					
Purpose:	**To demonstrate that a station will fail to select a slot when no slots are available which are compatible with the QoS parameters.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the first incremental broadcast reservation (incremental burst 1) from the station under test.
	6	record	RF	inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station B is < Q2a, b, c, d away from station under test) in slot beginning at time = inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B < Q2a, b, c, d away from the station under test, reporting B's position.
	9	send	RF	INCREM_BURST_b(16) (io:= 24; s = add_B) in slot beginning at time = inc_time + (4 x IO + 63 - 96) x 60/M1	Ib(k)	Send an incremental burst from station B < Q2a, b, c, d away reserving a series of 16 slots that conflict with the candidate range of the next incremental burst from the station under test.
	10	send	RF	INCREM_BURST_b(16) (io:= 24; s= add_C) in slot beginning at time = inc_time + (4 × IO + 63 - 96 + 16) × 60/M1	Ib(16)	Send an incremental burst from station C < Q2a, b, c, d away reserving a series of 16 slots that conflict with the candidate range of the next incremental burst from the station under test.
	11	await		time = inc_time + (4 x IO) x 60/M1		Wait for the slot reserved by the station under test for its next incremental broadcast reservation.
	12	verify	RF	No incremental broadcast reservation in slot beginning at time = inc_time + (4 x IO) x 60/M1		Verify that the reserved slot does not contain an incremental broadcast reservation (incremental burst 2) because the slot which it needed to reserve could not be selected.
	13	verify	VSS	VSS user informed that no slot could be selected for a further incremental broadcast reservation		Verify that the VSS user is informed that no slot could be selected for a further incremental broadcast reservation (incremental burst 3).
	14	rep 25		n:= 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	15	verify	RF	No transmission from station under test in slot beginning at time = inc_time + (4 x IO + 63 + n) x 60/M1		Verify that in the candidate range in which the station under test was attempting to reserve a slot, there is no incremental burst (incremental burst 3) from the station under test.
	16	endrep		n:= n + 1		
postamble	17	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	18	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:	**SlotSel_QoSGroup**					
Purpose:	**To demonstrate that a station will select a slot using a second group of QoS parameters when no slot has been selected by means of the first group.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts.
	4	send	VSS	INPUT Q2 SET 2	Q2 Set 2	Send to the station under test the Q2 Set 2 parameters in addition to the default Set 1, allowing it to use the less stringent Q2 Set 2 parameters when slot selection is unsuccessful with the first set.
test body	5	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	6	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the first incremental broadcast reservation from the station under test (incremental burst 1).
	7	record	RF	inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	8	record	RF	IO:= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	9	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station B is < Q2a, b, c, d away from station under test) in slot beginning at time = inc_time + 5 x 60/M1	Sa	Send a sync burst from a simulated station B < Q2a, b, c, d away from the station under test, reporting B's position.
	10	send	RF	INCREM_BURST_b(16) (io:= 24; s = add_B) in slot beginning at time = inc_time + (4 x IO + 63 - 96) x 60/M1	Ib(16)	Send an incremental burst from station B < Q2a, b, c, d away reserving a series of 16 slots that conflict with the candidate range of the next incremental burst from the station under test.
	11	send	RF	INCREM_BURST_b(16) (io:= 24; s= add_B) in slot beginning at time = inc_time + (4 × IO + 63 - 96 + 16) × 60/M1	Ib(16)	Send an incremental burst from station C < Q2a, b, c, d away reserving a series of 16 slots that conflict with the candidate range of the next incremental burst from the station under test.
	12	verify	RF	INCREM_BURST_a (s = add_A) in slot beginning at time = inc_time + (4 x IO) x 60/M1	Ia	Verify that the station under test makes use of the Q2 Set 2 parameters by being able to select a slot within the range of slots reserved by station B or station C, when it would not be able to do so without the Q2 Set 2. This slot therefore contains an incremental broadcast reservation (incremental burst 2) pointing to the selected slot.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	13	record	RF	inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	14	record	RF	IO2:= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	15	verify	RF	INCREM_BURST_a (s = add_A) in slot beginning at time = inc_time_2 + (4 x IO2) x 60/M1	Ia	Verify that the selected slot is used by the station under test to transmit a further incremental broadcast (incremental burst 3).
postamble	16	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	17	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default values.
Comments:						

Test Case Name:				**SlotSel_Exclusion**		
Purpose:	**To demonstrate that a station does not select a slot for transmission when the station is required to transmit in that slot on another channel.**					
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.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b (Q4:= 5; $TV11_{min}$:= 1; $TV11_{max}$:= 1; V11:= 1; V12:= (4/M1) × V11; f:= f0)	Sb	Set up a periodic stream of one-slot messages on channel with frequency f0 from the station under test. Q4 set to 5; equals number of slots in dither range available for selection. TV11 reservation hold timer set to force dither in next superframe. V11 set to 1. V12 set to give dither range of ±2.
	4	await	RF(f0)	SYNC_BURST_b (s= add_A)	Sb	
	5	record	RF(f0)	reserve_time:= time at the beginning of slot containing SYNC_BURST_b (s= add_A)	Sb	Define a reference time to measure relative times from during the test. This slot position will be used for the reserved slot after the station under test has dithered away from this slot.
	6	await		time = reserve_time + 60 - 50/M1× 60		Wait for reserve_time plus 1 superframe minus 50 slots.
	7	send	RF	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 50; lg:= 0; rcvr:= 0; f:= f1; r-mi:= 0; trmt:= 0; s:= add_G (address indicating source is a ground station); d:= add_A) in slot beginning at time = reserve_time + 60 - 50/M1× 60	Da	Send a directed burst from a simulated ground station G and with the transmit control (trmt) flag set to 0, requesting the station under test to transmit at a rate of 4 bursts per superframe for 5 superframes on frequency f1.
	8	rep 21		n:= 1		
	9	await	RF(f0)	SYNC_BURST_b (s= add_A)	Sb	
	10	record	RF(f0)	sync_time(n):= time at beginning of slot of n^{th} SYNC_BURST_b (s= add_A) diff_time:= sync_time(n) - (n - 1) × 60 - sync_time(1) ct_slot_diff(n):= diff_time × M1/60	Sb	Record the time of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each ct_slot and the ct_slot of the first burst and transpose to a common time reference. Convert time differences to slot differences.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	11	endrep		n:= n + 1		
	12	verify		MAX(ct_slot_diff(n)) - MIN(ct_slot_diff(n)) ≤ V12 × M1/V11		Verify distribution of slots is equal to or less than the candidate slot range.
	13	record		no_ct_slot_diff(m):= 0 for all m		Initialize array of variables to store frequency of occurrence of slots in each candidate slot position.
	14	rep 21		n:= 2		
	15	record		no_ct_slot_diff(ct_slot_diff(n)):= no_ct_slot_diff(ct_slot_diff(n)) + 1		Record the frequency of occurrence of slots in each candidate slot position.
	16	endrep		n:= n + 1		
	17	record		m_res_slot:= (reserve_time + 180 -sync_time(1)) × M1/60		Calculate relative slot difference between the reserved slot and the reference slot when transposed onto a common time reference.
	18	verify		no_ct_slot_diff(m_res_slot) = 0		Verify that no transmission is made on channel f0 in the slot which the station under test has been directed to transmit in by station G.
postamble	19	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	20	send	VSS	SET PARAMETERS (Q4:= 3; $TV11_{min}$:= 4; $TV11_{max}$:= 8; V11:= 1; V12:= 0.10)		Reset to default values.
	21	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Conflict_Periodic_A					
Purpose:	**To demonstrate that a station will continue to transmit a periodic stream without action in the event of a conflicting non-periodic transmission from another station.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS ($TV11_{min}$:= 15; $TV11_{max}$:= 16)		Set $TV11_{min}$ and $TV11_{max}$ to their maximum values. V11has default value of 1 burst per superframe.
test body	3	await	RF	First SYNC_BURST_c (s = add_A) following dither to a new slot in the superframe	Sc	Await the first sync burst following a dither to a new slot.
	4	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_c (s = add_A)	Sc	Define a reference time to measure relative times from during the test.
	5	await		time = sync_time + 50 x 60/M1		
	6	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 310 NM)) (position of station B is < Q2a, b, c, d away from station under test) in slot beginning at time = sync_time + 50 x 60/M1	Sa	Send a sync burst from a simulated station B < Q2a, b, c, d away from the station under test, reporting B's position (see note).
	7	await		time = sync_time + (M1 - 1 280) x 60/M1		
	8	send	RF	INCREM_BURST_a (io:= 240; s:= add_B) (position of station B is < Q2a, b, c, d away from station under test) in slot beginning at time = sync_time + (M1 - 960) x 60/M1	Ia	Send an incremental burst from the simulated station B < Q2a, b, c, d away from the station under test, reserving a slot that conflicts with the periodic stream (see note).
	9	rep 2		n:= 1		
	10	verify	RF	SYNC_BURST_c (s = add_A) in slot beginning at time = sync_time + n x 60	Sc	Verify that the periodic stream from the station under test continues without change.
	11	endrep		n:= n + 1		
postamble	12	send	VSS	SET PARAMETERS (V11:= 6; TV11min:= 4; $TV11_{max}$:= 8)		Reset to default values
Comments:						
NOTE: The value of the Q2a, b, c, d parameters used here is that specified within the Q2 Set 4 parameters shown in clauses 5.4.4.1.8, 5.4.4.1.9 and table 5.72 and defined in ICAO VDL4 Technical Manual [1], clause 1.5.5.1.4.						

Test Case Name:	**Conflict_Periodic_B**					
Purpose:	**To demonstrate that a station will dither a periodic stream to resolve a conflict with a periodic stream from another station.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS ($TV11_{min}$:= 15; $TV11_{max}$:= 16)		Set $TV11_{min}$ and $TV11_{max}$ to their maximum values. V11 has default value of 1 burst per superframe.
test body	3	await	RF	First SYNC_BURST_c (s = add_A) following dither to a new slot in the superframe	Sc	Await the first sync burst following a dither to a new slot.
	4	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_c (s = add_A)	Sc	Define a reference time to measure relative times from during the test.
	5	await		time = sync_time + 50 x 60/M1		
	6	send	RF	SYNC_BURST_a (pt:= 1; po:= -50; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 310 NM)) (position of station B is < Q2a, b, c, d away from station under test) in slot beginning at time = sync_time + 50 x 60/M1	Sa	Send a periodic reservation from a simulated station B < Q2a, b, c, d away from the station under test, with a periodic offset value reserving slots that conflict with the test station stream, and a periodic timer value such that the conflicting reservation starts 2 frames in the future (see note).
	7	await	RF	SYNC_BURST_c (s = add_A) in slot beginning at time = sync_time + 60	Sc	Wait for the sync burst in the superframe before the reservation conflict.
	8	record	RF	PO:= po of SYNC_BURST_c PT:= pt of SYNC_BURST_c	Sc	Record the value of the periodic timer indicating that the stream will dither in the next superframe, and that of the periodic offset identifying the slot to which it will dither.
	9	verify		PO ≠ 0 **AND** PO ≠ -128 **AND** PT = 0		Verify valid values for pt and po indicating that the station will dither to avoid conflict.
	10	await		time = sync_time + (2 x M1 + PO) x 60/M1		
	11	verify	RF	SYNC_BURST_c (s = add_A) in slot beginning at time = sync_time + (2 x M1 + PO) x 60/M1	Sc	Verify that the station under test has dithered the periodic stream to the announced slot in order to avoid conflict.
postamble	12	send	VSS	SET PARAMETERS (TV11min:= 4; $TV11_{max}$:= 8)		Reset to default values
Comments:						
NOTE: The value of the Q2a, b, c, d parameters used here is that specified within the Q2 Set 4 parameters shown in clauses 5.4.4.1.8, 5.4.4.1.9 and table 5.72 and defined in ICAO VDL4 Technical Manual [1], clause 1.5.5.1.4.						

Test Case Name:	Conflict_Periodic_C					
Purpose:	**To demonstrate that a station will move a periodic stream to a new location in the event of a conflict with a periodic stream from another station that does not allow the original stream to be dithered.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS ($TV11_{max}$:= 4; V12:= (2/M1) x V11)		Set $TV11_{max}$ to use dither every 4 superframes. Set V12 to minimum to give a dither range of ±1. V11 has default value of 1 burst per superframe.
test body	3	await	RF	First SYNC_BURST_c (s = add_A) following dither to a new slot in the superframe	Sc	Await the first sync burst following a dither to a new slot.
	4	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_c (s = add_A)	Sc	Define a reference time to measure relative times from during the test.
	5	await	RF	SYNC_BURST_c (s = add_A) in slot beginning at time = sync_time + 60	Sc	Wait for the second sync burst in the stream.
	6	record	RF	PO:= po of SYNC_BURST_c PT:= pt of SYNC_BURST_c	Sc	Record the periodic offset and periodic timer values. pt shall have a value of 2 (pt = TV11 - 1) here indicating continuing reservations in current slot for 2 more superframes before the stream dithers to a new slot as identified by po.
	7	await		time = sync_time + (M1 + 50) x 60/M1		
	8	send	RF	SYNC_BURST_a (pt:= 0; po:= -50; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 310 NM)) (position of station B is < Q2a, b, c, d away from station under test) in slot beginning at time = sync_time + (M1 + 50) x 60/M1	Sa	Send a periodic reservation from a simulated station B < Q2a, b, c, d away from the station under test, with a periodic offset value reserving slots that conflict with the test station stream, and a periodic timer value such that the conflicting reservation is in the next scheduled test station sync burst (third burst in stream).
	9	await	RF	SYNC_BURST_c (s = add_A)	Sc	Await the next burst from the station under test which should be the first burst of new stream.
	10	record	RF	new_sync_time:= time at the beginning of the slot containing SYNC_BURST_c (s = add_A)	Sc	
	11	verify		new_sync_time = sync_time + 60		Verify that the sync burst has not moved from its ct_slot.
	12	await	RF	SYNC_BURST_c (s= add_A)	Sc	Await the next burst from the station under test which should be the first burst of new stream.
	13	record	RF	new_sync_time:= time at the beginning of the slot containing SYNC_BURST_c (s= add_A)	Sc	

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	verify		new_sync_time ≠ sync_time + 120		Verify that the sync burst has moved from its ct_slot so as to avoid a conflict with the reserved slots.
postamble	15	send	VSS	SET PARAMETERS (V12:= 0,1; TV11min:= 4; $TV11_{max}$:= 8)		Reset to default values
Comments:						
NOTE: The value of the Q2a, b, c, d parameters used here is that specified within the Q2 Set 4 parameters shown in clauses 5.4.4.1.8, 5.4.4.1.9 and table 5.73 and defined in ICAO VDL4 Technical Manual [1], clause 1.5.5.1.4.						

Test Case Name:	**Conflict_Directed**					
Purpose:	**To demonstrate that a station will continue to transmit in a slot reserved for it by another station by a directed request, in the event of receiving a conflicting reservation from any other station.**					
Reference:	**1.3.6.5 a, 1.3.6.5 d**					
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.
test body	4	send	RF	DIR_REQ_a (or:= 0; dt:= 1; nr:= 1; do:= 50; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_B; d:= add_A)	Da	Send a burst from a station B requesting the station under test to transmit in the slot 50 slots after the start of the directed request reservation.
	5	record	RF	directed_time:= time at beginning of slot containing directed request reservation DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	6	await		time = directed_time + 26 × 60/M1		
	7	send	RF	INCREM_BURST_a (io:= 6; s:= add_E) in slot beginning at time = directed_time + 26 × 60/M1	Ia	Send an incremental burst from a station E, reserving a slot which conflicts with the slot reserved by the directed request. Send burst 24 slots before the slot reserved by the directed request.
	8	await		time = directed_time + 50 × 60/M1		
	9	verify	RF	slot contains DIR_SYNC_BURST_a (s:= add_A) from station under test	DSa	Verify that the station under test transmits in the slot reserved for it by the directed request from station B, in spite of the conflicting reservation from station E.
postamble	10	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	11	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Conflict_NoAction**					
Purpose:	**To demonstrate that a station will continue to transmit a periodic stream without action in the event of receiving a conflicting reservation such that the slot remains available.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SET PARAMETERS ($TV11_{min}$:= 15; $TV11_{max}$:= 16; V11:= 1)		Set $TV11_{min}$ and $TV11_{max}$ to their maximum values. Set V11 to 1 burst per superframe.
test body	3	await	RF	First SYNC_BURST_c (s = add_A) following dither to a new slot in the superframe	Sc	Await the first sync burst following a dither to a new slot.
	4	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_c (s = add_A)	Sc	Define a reference time to measure relative times from during the test.
	5	await		time = sync_time + 50 x 60/M1		
	6	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 370 NM)) (position of station B is > Q2b away from station under test) in slot beginning at time = sync_time + 50 x 60/M1	Sa	Send a sync burst from a simulated station B > Q2b away from the station under test, reporting B's position.
	7	await		time = sync_time + (M1 - 960) x 60/M1		
	8	send	RF	INCREM_BURST_a (io:= 240; s = add_B) in slot beginning at time = sync_time + (M1 - 960) x 60/M1	Ia	Send an incremental burst from a station B > Q2b away from the station under test, reserving a slot that conflicts with the periodic stream.
	9	rep 2		n:= 1		
	10	verify	RF	SYNC_BURST_c (s = add_A) in slot beginning at time = sync_time + n x 60	Sc	Verify that the periodic stream continues without change.
	11	endrep		n:= n + 1		
postamble	12	send	VSS	SET PARAMETERS ($TV11_{min}$:= 4; $TV11_{max}$:= 8; V11:= 6)		Reset to default values.
Comments:						
NOTE: The value of the Q2b parameter used here is that specified within the Q2 Set 4 parameters shown in clauses 5.4.4.1.8, 5.4.4.1.9 and table 5.73 and defined in ICAO VDL4 Technical Manual [1], clause 1.5.5.1.4.						

Test Case Name:	Conflict_Incremental					
Purpose:	**To demonstrate that a station will not transmit in a slot previously reserved by an incremental broadcast reservation in the event of receiving a conflicting reservation, and will make the broadcast in an alternative slot by random access.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots (io:= 300)	Ia	Set up a series of incremental broadcasts from the station under test to transmit every 1 200 slots.
	4	await	RF	INCREM_BURST_a (io= 300; s = add_A)	Ia	Wait for first incremental burst.
	5	record	RF	sync_time:= time at the beginning of slot containing INCREM_BURST_a (io= 300; s = add_A)	Ia	Define a reference time to measure relative times from during the test.
	6	await		time = sync_time + 300 x 60/M1		
	7	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; s = add_B) in slot beginning at time:= sync_time + 300 x 60/M1	Sb	Send a sync burst from a station B with a reservation conflicting with a future incremental broadcast from the station under test.
	8	await	RF	INCREM_BURST_a (io:= 300; s = add_B) in slot beginning at time = sync_time + 3 600 x 60/M1	Ia	
	9	verify verify	RF RF	next INCREM_BURST_a (io:= 300; s = add_B) occurs in or after slot beginning at time = sync_time + 4 350 x 60/M1 and in or before slot beginning at time = sync_time + 5 250 x 60/M1 **AND** slot beginning at time = sync_time + 4 800 x 60/M1 contains SYNC_BURST_b (po:= 0; pt:= 2; s = add_B)	Ia, Sb	Verify that the incremental broadcast is moved to a new slot to avoid the conflict (using the random access protocol).
postamble	10	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Conflict_BND**					
Purpose:	**To demonstrate that a station which has made a BND reservation will not transmit in the reserved slot in the event of a conflicting reservation made by a periodic broadcast**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	rep 10		n:= 1		Repeat the test 10 times.
	2	do		switch on VDL4 transceiver		
	3	verify	Selftest	successful VDL4 transceiver selftest		Verify that the VDL4 transceiver has successfully passed power-up test.
	4	do		NETWORK ENTRY BY DELAYED BND		Ensure transceiver is set to perform network entry by delayed BND transmission as opposed to other means.
test body	5	await	RF	BND_DELAYED_a (s:= add_A)	BDa	Wait for a delayed transmission containing a BND reservation from the station under test.
	6	record	RF	bnd_time:= time at beginning of slot occupied by BND_DELAYED_a (s= add_A)	BDa	
	7	record	RF	ND:= nd contained in BND_DELAYED_a (s= add_A)	BDa	Record the nd value indicating the slot pointed to by the BND reservation.
	8	await		time = bnd_time + (M1 - 128 - (4 × ND) - 40) × 60/M1		Wait until 40 slots before the BND reservation.
	9	send	RF	INCREM_BURST_a (io:=10; s:= add_B) in slot beginning at time = bnd_time + (M1 - 128 - (4 × ND) - 40) × 60/M1	Ia	Send an incremental burst from a simulated station B reserving a slot that conflicts with the BND reservation.
	10	await		time = bnd_time + (M1 - 128 - (4 × ND)) × 60/M1		Wait for slot reserved by the BND reservation.
	11	verify	RF	No transmission by station under test in slot beginning at time = bnd_time + (M1 - 128 - (4 × ND)) × 60/M1		Verify that the station under test does not transmit in the reserved slot.
postamble	12	do		switch off VDL4 transceiver		Switch off the transceiver prior to repeating the test.
	13	endrep		n:= n + 1		End loop.
Comments: Network entry by delayed BND is not mandated by ICAO standards and nor by the present document. Step 4 is provided to ensure that this means of net entry is selected in preference to other means, such as waiting for one minute. If the transceiver under test does not support network entry by delayed BND, then this test does not apply.						

Test Case Name:	**Conflict_Priority**					
Purpose:	**To demonstrate that a station required to transmit in the same slot by conflicting requests will transmit the response of highest priority.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	rep 10		n:= 1		
	4	send	RF	UNI_BURST_d (ro:= 100; lg:= 0; sdf:= 0; pr:= 1; r mi:= xxxxx10; s:= add_B; d:= add_A)	Ud	Send a unicast request burst from a simulated station B to the station under test with priority pr = 1 carrying a general request for a sync burst.
	5	record	RF	uni_start:= time at beginning of slot containing UNI_BURST_d	Ud	Record the time the unicast burst was sent.
	6	send	RF	UNI_BURST_d (ro:= 97; lg:= 0; sdf:= 0; pr:= 2; r mi:= xxxxx10; s:= add_C; d:= add_A) in slot beginning at time = uni_start + 3	Ud	Send a unicast request burst from a simulated station C to the station under test with priority pr = 1 carrying a general request for a sync burst. The transmission reserves the same slot for a response as the transmission from station B.
	7	verify	RF	SYNC_BURST_m (s:= add_A; d:= add_C) in slot beginning at time = uni_start + 100	Sm	Verify that the station under test responds to station C in the reserved slot with a sync burst with the response reservation address set to the address of station C.
	8	endrep		n:= n + 1		
postamble	9	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments: Certain stations, such as those with low power (Type B) transmitters, will not support a general request for a sync burst using a unicast reservation. For such stations, this test is inapplicable.						

Test Case Name:	**Conflict_FirstRequest**					
Purpose:	**To demonstrate that a station required to transmit in the same slot by conflicting requests of equal priority will transmit the response to the first request.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	rep 10		n:= 1		
	4	send	RF	UNI_BURST_d (ro:= 100; lg:= 0; sdf:= 0; pr:= 2; r mi:= xxxxx10; s:= add_B; d:= add_A)	Ud	Send a unicast burst from a simulated station B to the station under test, with sdf = 0 and priority pr = 2, carrying a general request for a sync burst.
	5	record	RF	uni_start:= time at beginning of slot containing UNI_BURST_d	Ud	Record the time the compressed frame burst was sent.
	6	send	RF	UNI_BURST_d (ro:= 97; lg:= 0; sdf:= 0; pr:= 2; r-mi:= xxxxx10; s:= add_C; d:= add_A) in slot beginning at time = uni_start + 3 x 60/M1	Ud	Send a unicast burst from a simulated station C to the station under test, with sdf = 0 and priority pr = 2, carrying a general request for a sync burst. The transmission reserves the same slot for a response as the transmission from station B.
	7	verify	RF	SYNC_BURST_m (s:= add_A; d:= add_B) in slot beginning at time = uni_start + 101 x 60/M1	Sm	Verify that the station under test responds to station B in the reserved slot with a sync burst with the response reservation address set to the address of station B.
	8	endrep		n:= n + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
postamble	9	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments: Certain stations, such as those with low power (Type B) transmitters, will not support a general request for a sync burst using a unicast reservation. For such stations, this test is inapplicable.						

Test Case Name:	**Conflict_Channel_Priority**					
Purpose:	**To demonstrate that a station required to transmit in the same slot on different channels by conflicting requests will transmit the response of highest priority.**					
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.
test body	3	rep 10		n:= 1		
	4	send	RF (GSC1)	UNI_BURST_d (ro:= 100; lg:= 0; sdf:= 0; pr:= 1; r-mi:= xxxxx10; s:= add_B; d:= add_A)	Ud	Send a unicast request burst from a simulated station B to the station under test on GSC 1 with priority pr = 1 carrying a general request for a sync burst.
	5	record	RF (GSC1)	uni_start:= time at beginning of slot containing UNI_BURST_d	Ud	Record the time the unicast burst was sent on GSC 1.
	6	send	RF (GSC2)	UNI_BURST_d (ro:= 97; lg:= 0; sdf:= 0; pr:= 2; r-mi:= xxxxx10; s:= add_C; d:= add_A) in slot beginning at time = uni_start + 3	Ud	Send a unicast request burst from a simulated station C to the station under test on GSC 2 with priority pr = 2 carrying a general request for a sync burst.. The transmission reserves the same slot for a response as the transmission from station B.
	7	verify	RF (GSC2)	SYNC_BURST_m (s:= add_A; d:= add_C) in slot beginning at time = uni_start + 100	Sm	Verify that the station under test responds to station C in the reserved slot on GSC 2 with a sync burst with the response reservation address set to the address of station C.
	8	endrep		n:= n + 1		
postamble	9	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments: Certain stations, such as those with low power (Type B) transmitters, will not support a general request for a sync burst using a unicast reservation. For such stations, this test is inapplicable.						

Test Case Name:	Conflict_Channel_Ground_A					
Purpose:	**To demonstrate that when a station is requested to make a sync burst transmission in the same slot on different channels, the request from the ground station takes precedence.**					
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.
test body	4	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 2; nr:= 1; do:= 100; lg:= 0; rcvr:= 11; f:= f1; trmt:= 1; r-mi:= 0; s:= add_B; d:= add_A)	Da	Send a burst from a simulated mobile station B requesting the station under test to transmit in the slot 100 slots after the start of the directed request reservation on channel f1, and in the same slot for a further 2 superframes.
	5	record	RF (GSC1)	directed_time:= time at beginning of slot containing directed request reservation DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	6	await		time:= directed_time + 10 × 60/M1		
	7	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 0; nr:= 1; do:= 90; lg:= 0; rcvr:= 11; f:= f2; trmt:= 1; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a burst from a simulated ground station G requesting the station under test to transmit in the same slot but on channel f2 for one superframe.
	8	await		time:= directed_time + 100 × 60/M1		
	9	verify	RF (f2)	DIR_SYNC_BURST_a (s:= add_A) in slot beginning at time:= directed_time + 100 × 60/M1	DSa	Verify that the station under test transmits in the slot reserved for it by the directed request from ground station G, in spite of the conflicting directed request from mobile station B.
	10	await		time:= directed_time + 60 + 100 × 60/M1		
	11	verify	RF (f1)	DIR_SYNC_BURST_a (s:= add_A) in slot beginning at time:= directed_time + 60 + 100 × 60/M1	DSa	Verify that the station under test transmits in the same slot on channel f1 in the following superframe according to the request from mobile B.
postamble	12	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	13	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Conflict_Channel_Ground_B**					
Purpose:	**To demonstrate that when a station is requested to make a sync burst transmission in the same slot on different channels by more than one ground station, the latest request takes precedence.**					
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.
test body	4	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 2; nr:= 1; do:= 100; lg:= 0; rcvr:= 11; f:= f1; trmt:= 1; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a burst from a simulated ground station G requesting the station under test to transmit in the slot 100 slots after the start of the directed request reservation on channel f1, and in the same slot for a further 2 superframes.
	5	record	RF (GSC1)	directed_time:= time at beginning of slot containing directed request reservation DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	6	await		time:= directed_time + 10 × 60/M1		
	7	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 0; nr:= 1; do:= 90; lg:= 0; rcvr:= 11; f:= f2; trmt:= 1; r-mi:= 0; s:= add_H (address indicating source is a ground station); d:= add_A)	Da	Send a burst from a simulated ground station H requesting the station under test to transmit in the same slot but on channel f2 for one superframe.
	8	await		time:= directed_time + 100 × 60/M1		
	9	verify	RF (f2)	DIR_SYNC_BURST_a (s:= add_A) in slot beginning at time:= directed_time + 100 × 60/M1	DSa	Verify that the station under test transmits in the slot reserved for it by the latest directed request from ground station H, in spite of the conflicting directed request from ground station G.
	10	await		time:= directed_time + 60 + 100 × 60/M1		
	11	verify	RF (f1)	DIR_SYNC_BURST_a (s:= add_A) in slot beginning at time:= directed_time + 60 + 100 × 60/M1	DSa	Verify that the station under test transmits in the same slot on channel f1 in the following superframe according to the request from ground station G.
postamble	12	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	13	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Conflict_Channel_FirstRequest**					
Purpose:	**To demonstrate that when a station is requested to make a sync burst transmission in the same slot on different channels by more than one ground station, the latest request takes precedence.**					
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.
test body	4	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 2; nr:= 1; do:= 100; lg:= 0; rcvr:= 11; f:= f1; trmt:= 1; r-mi:= 0; s:= add_B; d:= add_A)	Da	Send a burst from a simulated mobile station B requesting the station under test to transmit in the slot 100 slots after the start of the directed request reservation on channel f1, and in the same slot for a further 2 superframes.
	5	record	RF (GSC1)	directed_time:= time at beginning of slot containing directed request reservation DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	6	await		time:= directed_time + 10 × 60/M1		
	7	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 0; nr:= 1; do:= 90; lg:= 0; rcvr:= 11; f:= f2; trmt:= 1; r-mi:= 0; s:= add_C; d:= add_A)	Da	Send a burst from a simulated mobile station C requesting the station under test to transmit in the same slot but on channel f2 for one superframe.
	8	await		time:= directed_time + 100 × 60/M1		
	9	verify	RF (f2)	DIR_SYNC_BURST_a (s:= add_A) in slot beginning at time:= directed_time + 100 × 60/M1	DSa	Verify that the station under test transmits in the slot reserved for it by the latest directed request from mobile station C, in spite of the conflicting directed request from mobile station B.
	10	await		time:= directed_time + 60 + 100 × 60/M1		
	11	verify	RF (f1)	DIR_SYNC_BURST_a (s:= add_A) in slot beginning at time:= directed_time + 60 + 100 × 60/M1	DSa	Verify that the station under test transmits in the same slot on channel f1 in the following superframe according to the request from mobile station B.
postamble	12	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	13	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Slot_Boundary**					
Purpose:	**To demonstrate that a transmission from the station complies with timing performance requirements at the slot boundary.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
	4	do		MEASURE NOISE FLOOR		Measure the channel idle power level in order to estimate the noise floor.
test body	5	macro		M_RAND_ACC (sf:= 1)		Queue random access transmissions over 1 superframes.
	6	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	7	rep 10		n:= 1		
	8	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the next random access transmission.
	9	await	RF	500 μs before end of slot containing RAND_ACC_DATA_a (s = add_A) (measured from the test equipment's UTC slot start time)	Ra	Wait until 500 μs before the end of the slot.
	10	record	RF	start_time:= time 500 μs before end of next slot containing RAND_ACC_DATA_a (s = add_A)	Ra	
	11	rep 20 000		p:= 0		Define a sequence of points at which to measure the transmission amplitude.
	12	record record	RF	Measure transmission amplitude trans_amp at time:= start_time + p x 10^{-7} Calculate transmission power trans_power(trans_amp)		Measure the transmission amplitude at each point. Calculate the transmission power at each point with respect to noise floor.
	13	endrep		p:= p + 1		
	14	record		steady_power:= trans_power averaged over last 4 000 points		Measure the steady state channel busy power level.
	15	verify verify		trans_amp = 0 before nominal slot start time (measured from the test equipment's UTC slot start time) **AND** trans_power ≥ 0,9 x steady_power at 833,3 ± 5 μs after the nominal slot start time (measured from the test equipment's UTC slot start time)		Verify that the transmission does not begin before the nominal start of the slot, and that 16 symbol periods (833,3 ± 5 μs) after the nominal start of the slot, the transmitter power level has increased to at least 90 % of the steady state channel busy power level.
	16	endrep		n:= n + 1		
postamble	17	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Slot_Delayed					
Purpose:	**To demonstrate that a delayed transmission from the station complies with timing performance requirements.**					
Reference:	**1.2.5 c, 1.3.7.2.2 b**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	rep 10		n:= 1		Repeat the test n times.
	2	do		switch on VDL4 transceiver		
	3	verify	Selftest	successful VDL4 transceiver selftest		Verify that the VDL4 transceiver has successfully passed power-up test.
	4	do		NETWORK ENTRY BY DELAYED PLEA		Ensure transceiver is set to enable network entry by delayed plea transmission, if supported, as opposed to other means.
	5	send	VSS	SET PARAMETERS (CG1_plea:= 1)		Set the minimum number of stations which must be identified before a plea can be issued to 1.
Test body	6	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 10 NM))	Sa	Send a sync burst from a simulated station B > Q2a away from the station under test, reporting B's position.
	7	await	RF	PLEA_a (s= add_A; d= add_B)	Pa	Wait for the station under test to issue a delayed plea transmission to station B, ignoring any BND transmissions.
	8	record	RF	t:= time at which transmission begins in the slot containing PLEA_a, measured from the test equipment's UTC slot start time	Pa	Record the time after the start of the nominal slot boundary that the transmission begins.
	9	verify		t = 4 000 ± 5 microseconds		Verify that the transmission begins 4 000 ± 5 microseconds after the nominal slot boundary.
postamble	10	do		switch off VDL4 transceiver		Switch off the transceiver prior to repeating the test.
	11	endrep		n:= n + 1		End loop.
Comments: Network entry by delayed plea transmission is not mandated by ICAO standards, and nor by the present document. Step 4 is provided to ensure that this means of net entry is selected in preference to other means, such as waiting for one minute. In the event that the transceiver under test does not support network entry by delayed plea, then this test does not apply.						

Test Case Name:	**Rand_Busy**					
Purpose:	**To demonstrate that a station will not make a random access transmission in a slot perceived to be busy at the start of the slot (e.g. a transmission which extends beyond the guard time).**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_b (pt:= 1; po:= 0; s:= add_B)	Sb	Send a sync burst (burst length 1) from a simulated station B reserving the same transmission slot in the next superframe, but thereafter terminating the stream.
	5	record	RF	periodic_start:= time at beginning of slot containing the sync burst		Provides a reference time for the next burst from station B.
	6	macro		M_RAND_ACC (sf:= 1)		Queue random access transmissions over 1 superframe.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	
	8	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s = add_A)	Ra	Define a reference time to measure relative times from during the test.
	9	repx		n:= 1		
	10	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = start_time + n x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots up to the reserved slot.
	11	until		time = periodic_start + (M1 - 1) x 60/M1 in previous step; n:= n + 1		
	12	await		time = periodic_start + 60		Wait for the start of the next superframe.
	13	send	RF	SYNC_BURST_e (pt:= 3; po:= 0; s = add_B) in slot beginning at time:= periodic_start + 60	Se	Send a burst with pt = 3 and po = 0 from station B extending over one slot boundary into the following slot.
	14	repx		n:= 1		
	15	 verify verify	 RF RF	**IF** n = 1 **THEN** no transmission from station under test present in slot beginning at time = periodic_start + (n + M1) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = add_A) in slot beginning at time = periodic_start + (n + M1) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the slot following the reserved slot.
	16	until		time = start_time + 60; n:= n + 1		Ends the loop 1 minute after the first random access transmission was sent, i.e. verification takes place over 1 superframe + 1 slot.
postamble	17	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Rand_Congestion**					
Purpose:	**To demonstrate that the VSS User is informed if a request to make a random transmission is not successful within TM2 slots.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	RF	SYNC_BURST_d(16) (pt:= 3; po:= 0; s= add_B)	Sd(16)	Send a sync burst from a simulated station B extending over 16 slots.
	4	record	RF	sync_time:= time at beginning of slot containing SYNC_BURST_d(16)	Sd(16)	Record the time the sync burst was transmitted.
	5	await		time:= sync_time + 16 × 60/M1		
	6	send	RF	SYNC_BURST_d(16) (pt:= 3; po:= 0; s= add_C) in slot beginning at time:= sync_time + 16 × 60/M1	Sd(16)	Send a sync burst from a simulated station C extending over 16 slots.
	7	await		time:= sync_time + 60 + 1 × 60/M1		
	8	send	VSS	RANDOM TRANSMISSION request to transmit RAND_ACC_DATA_a (TM2:= 25)		Send (VSS) a request for a random transmission (with TM2 = 25 slots).
	9	verify	VSS	message sent to vss user notifying congestion		Verify (VSS) that congestion is notified.
postamble	10	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Rand_Persistence**					
Purpose:	**To demonstrate that a random transmission is made with probability p.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	rep 2		m:= {104/256, 50/256}; exp(104/256, k):= {40,63, 24,12, 14,32, 8,50, 5,05, 7,38} for k = 1 to 6; exp(48/256, k):= {18,75, 15,23, 12,38, 10,06, 8,17, 35,41} for k = 1 to 6		m defines the two values to be used for the probability of transmission for a random access attempt. exp(m, k) gives the expected numbers of actual transmissions in each of the five slots following the request for random access transmission.
	4	send	VSS	SET PARAMETERS (p:= m)		Set the probability of transmission p for a random access attempt.
	5	record		no_slot(k):= 0 for k:= 1 to 6		Initialize to zero the number of transmissions in each slot position after the request for random transmission.
	6	rep 100		n:= 1		
	7	send	VSS	RANDOM TRANSMISSION request to transmit RAND_ACC_DATA_a		Send (VSS) a request for a random transmission.
	8	record	VSS	req_time:= time of first slot boundary after RANDOM TRANSMISSION request is sent		Record the time of the first slot boundary after the request for random transmission is sent.
	9	rep 5		x:= 1; inslot:= FALSE		
	10	record	RF	**IF** transmission present in slot beginning at time = req_time + (x - 1) x 60/M1 **THEN** no_slot(x):= no_slot(x) + 1 **AND** inslot:= TRUE		
	11	endrep		x:= x + 1		
	12	record		**IF** inslot:= FALSE **THEN** no_slot(6):= no_slot(6) + 1		
	13	await		req_time + 50 x 60/M1		
	14	endrep		n:= n + 1		
	15	rep 6		k:= 1; chi_squared:= 0		Initialize value of k to correspond to the first slot after the requests. Initialize chi_squared.
	16	record		chi_squared:= chi_squared + (no_slot(k) - exp (m, k))^2 / exp(m, k)		The distribution is tested for consistency with the value chosen for the probability of transmission p, by calculating the value of chi_squared.
	17	endrep		k:= k + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	18	verify		chi_squared < 13,4		Value of chi_squared shall be less than 13,4 for confidence that the distribution is consistent with the value chosen for p (5 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
	19	endrep		next m		
postamble	20	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	21	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Rand_MaxAttempts**					
Purpose:	**To demonstrate that the station will authorize a random transmission as soon as the channel is available after VS3 unsuccessful attempts**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
	3	send	VSS	SET PARAMETERS (p:= 5/256)		Set probability of transmission on access to near minimum value.
test body	4	rep 2		m:= {4, 9}; exp(4, k):= {10, 9,8, 9,61, 9,42, 473,16} for k = 1 to 5; exp(9, k):= {10, 9,8, 9,61, 9,42, 9,24, 9,06, 8,88, 8,71, 8,54, 428,72} for k = 1 to 10		m defines the two values to be used for the maximum number of access attempts VS3. Exp(m, k) gives the expected numbers of actual transmissions in each of the m + 1 slots following the request for random access transmission.
	5	send	VSS	RANDOM TRANSMISSION request to transmit RAND_ACC_DATA_a (VS3:= m)		Send (VSS) a request for a random transmission (with TM2 = 20 slots).
	6	record		no_slot(k):= 0 for k:= 1 to 5		Initialize to zero the number of transmissions in each slot position after the request for random transmission.
	7	rep 512		n:= 1		
	8	send	VSS	RANDOM TRANSMISSION request to transmit RAND_ACC_DATA_a		Send (VSS) a request for a random transmission.
	9	record	VSS	req_time:= time of first slot boundary after RANDOM TRANSMISSION request is sent		Record the time of the first slot boundary after the request for random transmission is sent.
	10	rep 12		x:= 1		
	11	record	RF	**IF** transmission present in slot beginning at time = req_time + (x - 1) x 60/M1 **THEN** no_slot(x):= no_slot(x) + 1		
	12	endrep		x:= x + 1		
	13	endrep		n:= n + 1		
	14	repx		k:= 1; chi_squared:= 0		Initialize value of k to correspond to the first slot after the requests. Initialize chi_squared.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	15	record		chi_squared:= chi_squared + (no_slot(k) - exp(m, k))^2 / exp(m, k)		The distribution is tested for consistency with the value chosen for the maximum number of access attempts VS3, by calculating the value of chi_squared.
	16	until		k:= m + 1; k:= k + 1		
	17	 verify verify		**IF** m:= 4 **THEN** chi_squared < 11,7 **ELSE** chi_squared < 19,7		When m = 4, the value of chi_squared shall be less than 11,7 for confidence that the distribution is consistent with the value chosen for VS3 (4 degrees of freedom). When m = 9, the value of chi_squared shall be less than 19,7 for confidence that the distribution is consistent with the value chosen for VS3 (9 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
	18	endrep		next m		
postamble	19	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	20	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Rand_Priority**					
Purpose:	**To demonstrate that bursts queued for transmission by random access are transmitted in order of priority.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_b (po:= 0; pt:= 1; s = add_B)	Sb	Send a sync burst from a simulated station B reserving a slot in the next superframe.
	5	record	RF	sync_time:= time at beginning of slot containing SYNC_BURST_b	Sb	
	6	rep 49		p:= 1		
	7	send	RF	SYNC_BURST_b (po:= 0; pt:= 1; s = add_B) in slot beginning at time = sync_time + p x 60/M1	Sb	Send a sync burst from station B in each slot, each one reserving a slot in the next superframe.
	8	endrep		p:= p + 1		
	9	await		time:= sync_time + 60		
	10	send	VSS	SET PARAMETERS (Q1:= 0)		Set priority of transmissions to low.
	11	rep 5		n:= 1		Maintains transmissions over sf superframes.
	12	queue	VSS	DATA_a(m)	Da(m)	Send packets of data (DATA_a) to the station under test for subsequent transmission by the random access protocol. Packets consist of repeating 10101010 bit sequence over m bits.
	13	endrep		n:= n + 1		Send slots random access transmissions.
	14	send	VSS	SET PARAMETERS (Q1:= 1)		Set priority of transmissions to medium.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	15	rep 5		n:= 1		Maintains transmissions over sf superframes.
	16	queue	VSS	DATA_a(m)	Da(m)	Send packets of data (DATA_a) to the station under test for subsequent transmission by the random access protocol.
	17	endrep		n:= n + 1		Send slots random access transmissions.
	18	send	VSS	SET PARAMETERS (Q1:= 1)		Set priority of transmissions to high.
	19	rep 5		n:= 1		Maintains transmissions over sf superframes.
	20	queue	VSS	DATA_a(m)	Da(m)	Send packets of data (DATA_a) to the station under test for subsequent transmission by the random access protocol.
	21	endrep		n:= n + 1		Send slots random access transmissions.
	22	await		time:= sync_time + 60 + 48 x 60/M1		
	23	rep 15		n:= 1		
	24	 verify verify verify	 RF RF RF	**IF** n = {1,2,3,4,5} **THEN** RAND_ACC_DATA_a (s = add_A) of high priority transmitted in slot beginning at time = start_time + 60 + (48 + n) x 60/M1 **ELSE IF** n = {6,7,8,9,10} **THEN** RAND_ACC_DATA_a (s = add_A) of medium priority transmitted in slot beginning at time = start_time + 60 + (48 + n) x 60/M1 **ELSE IF** n = {11,12,13,14,15} **THEN** RAND_ACC_DATA_a (s = add_A) of low priority transmitted in slot beginning at time = start_time + 60 + (48 + n) x 60/M1	Ra	Verify that random access bursts are transmitted in order of priority (highest first).
	25	endrep		n:= n + 1		
postamble	26	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	27	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
Comments:						

Test Case Name:	**Rand_TM2Reset**					
Purpose:	**To demonstrate that timer TM2 is reset following a successful random transmission when a further burst is queued for transmission.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
	3	send	VSS	SET PARAMETERS (TM2:= 20; p:= 1)		Ensure 100 % chance of transmission on access.
test body	4	send	RF	SYNC_BURST_k(12) (pt:= 3; po:= 0; a/d:= 0; s:= add_B, address indicating source is a ground station; lat:= CPR_LAT(0); lon:= CPR_LON (E 100 NM)) (position of station B is < Q2a, b away from station under test)	Sk(12)	Send a sync burst 12 slots in length from a simulated station B <Q2a, b away from the station under test, reserving slots for a burst >TM2/2 slots long.
	5	record	RF	sync_time:= time at beginning of slot containing SYNC_BURST_k(12)	Sk(12)	
	6	await		time:= sync_time + 13 x 60/M1		Leave one slot between the two sync bursts.
	7	send	RF	SYNC_BURST_k(12) (pt:= 3; po:= 0; a/d:= 0; s:= add_B, address indicating source is a ground station; lat:= CPR_LAT(0); lon:= CPR_LON(E 100 NM)) (position of station B is < Q2a, b away from station under test) in slot beginning at time:= sync_time + 13 x 60/M1	Sk(12)	Send a second sync burst 12 slots in length from station B <Q2a, b away from the station under test, reserving slots for a burst >TM2/2 slots long.
	8	await		time:= sync_time + 60		
	9	macro		M_RAND_ACC_SL (slots:= 2) at time = sync_time + 60		Queue random access transmissions over 2 slots.
	10	await		time:= sync_time + 60 + 12 x 60/M1		
	11	verify	RF	RAND_ACC_DATA_a (s = add_A) transmitted in slot beginning at time = sync_time + 60 + 12 x 60/M1	Ra	Verify that the first random access transmission is made in the vacant slot following the first burst from simulated station B.
	12	await		time:= sync_time + 60 + 25 x 60/M1		
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	13	verify	RF	RAND_ACC_DATA_a (s = add_A) transmitted in slot beginning at time = sync_time + 60 + 25 x 60/M1	Ra	Verify that the second random access transmission is made in the next vacant slot.
postamble	14	verify	VSS	No notification of congestion has been delivered.		Verify that no notification of congestion is delivered to the VSS user.
	15	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	16	send	VSS	SET PARAMETERS (TM2:= 1 500; p:= 64/256)		Reset to default values.
Comments:						

Test Case Name:	**Rand_TM2Clear**					
Purpose:	**To demonstrate that timer TM2 is cleared following a successful random transmission when no further bursts are queued for transmission.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
	3	send	VSS	SET PARAMETERS (TM2:= 20; p:= 1)		Ensure 100 % chance of transmission on access.
test body	4	send	RF	SYNC_BURST_k(12) (pt:= 3; po:= 0; a/d:= 0; s:= add_B, address indicating source is a ground station; lat:= CPR_LAT(0); lon:= CPR_LON (E 100 NM)) (position of station B is < Q2a, b away from station under test)	Sk(12)	Send a sync burst 12 slots in length from a simulated station B <Q2a, b away from the station under test, reserving slots for a burst >TM2/2 slots long.
	5	record	RF	sync_time:= time at beginning of slot containing SYNC_BURST_k(12)	Sk(12)	
	6	await		time:= sync_time + 13 x 60/M1		Leave one slot between the two sync bursts.
	7	send	RF	SYNC_BURST_k(12) (pt:= 3; po:= 0; a/d:= 0; s:= add_B, address indicating source is a ground station; lat:= CPR_LAT(0); lon:= CPR_LON (E 100 NM)) (position of station B is < Q2a, b away from station under test) in slot beginning at time:= sync_time + 13 x 60/M1	Sk(12)	Send a second sync burst 12 slots in length from station B <Q2a, b away from the station under test, reserving slots for a burst >TM2/2 slots long.
	8	await		time:= sync_time + 60		
	10	macro		M_RAND_ACC_SL (slots:= 1) at time = sync_time + 60		Queue a random access transmission over 1 slot.
	11	await		time:= sync_time + 60 + 12 x 60/M1		
	12	verify	RF	RAND_ACC_DATA_a (s = add_A) transmitted in slot beginning at time = sync_time + 60 + 12 x 60/M1	Ra	Verify that the random access transmission is made in the vacant slot following the first burst from simulated station B.
	13	await		time:= sync_time + 60 + 13 x 60/M1		
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	15	macro		M_RAND_ACC (slots:= 1) at time = sync_time + 60 + 13 x 60/M1		Queue a second random access transmission over 1 slot.
	16	verify	RF	RAND_ACC_DATA_a (s = add_A) transmitted in slot beginning at time = sync_time + 60 + 25 x 60/M1	Ra	Verify that the second random access transmission is made in the next vacant slot.
	17	verify	VSS	No notification of congestion has been delivered.		Verify that no notification of congestion is delivered to the VSS user.
postamble	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	19	send	VSS	SET PARAMETERS (TM2:= 1 500; p:= 64/256)		Reset to default values.
Comments:						

Test Case Name:	Rand_VS3Clear					
Purpose:	**To demonstrate that if a request to make a random transmission is not successful within TM2 slots then the VS3 counter is cleared and a transmission is made as soon as the channel becomes available.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
	3	send	VSS	SET PARAMETERS (TM2:= 25; p:= 1; VS3:= 1)		Ensure 100 % chance of transmission on access.
test body	4	send	RF	SYNC_BURST_k(16) (pt:= 3; po:= 0; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON (E 100 NM)) (position of station B is < Q2a, b away from station under test)	Sk(16)	Send a sync burst 16 slots in length from a simulated station B.
	5	record	RF	sync_time:= time at beginning of slot containing SYNC_BURST_k(16)	Sk(16)	
	6	await		time:= sync_time + 16 × 60/M1		
	7	send	RF	SYNC_BURST_k(16) (pt:= 3; po:= 0; a/d:= 0; s:= add_C; lat:= CPR_LAT(0); lon:= CPR_LON(E 105 NM)) (position of station B is < Q2a, b away from station under test) in slot beginning at time:= sync_time + 16 × 60/M1	Sk(16)	Send a sync burst 16 slots in length from a simulated station C.
	8	await		time:= sync_time + 60 + 1 × 60/M1		
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	9	macro		M_RAND_ACC_SL (slots:= 1) at time = sync_time + 60 + 1 × 60/M1		Queue a random access transmission over 1 slot.
	10	await		time:= sync_time + 60 + 32 x 60/M1		
	11	verify	RF	RAND_ACC_DATA_a (s:= add_A) transmitted in slot beginning at time:= sync_time + 60 + 32 x 60/M1		Verify that the random access transmission is made in the first available slot.
postamble	12	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	13	send	VSS	SET PARAMETERS (TM2:= 1 500; p:= 64/256; VS3:= 24)		Reset to default values.
Comments:						

Test Case Name:	**Rand_Availability**					
Purpose:	**To demonstrate that a station makes random access attempts in slots available only at levels 0 to 2.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
	3	send	VSS	SET PARAMETERS (TM2:= 20; p:= 1; VS3:= 5)		Ensure 100 % chance of transmission on access.
	4	send	VSS	INPUT Q2 SET 3	Q2 Set 3	Send to the station under test the VSS User defined Q2 Set 3 parameters.
test body	5	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 140 NM)) (position of station B is > Q2c and < Q2a, b away from station under test)	Sa	Send a sync burst from a simulated station B > Q2c and < Q2a, b away from the station under test.
	6	record	RF	sync_time:= time at beginning of slot containing SYNC_BURST_a	Sa	
	7	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_C; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station C is > Q2a, b away from station under test) in slot beginning at time = sync_time + 1 x 60/M1	Sa	Send a sync burst from a simulated station C > Q2a, b away from the station under test.
	8	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 170 NM)) (position of station D is > Q2a, b away from station under test and such that a transmission from C to D is CCI protected) in slot beginning at time = sync_time + 50 x 60/M1	Sa	Send a sync burst from a simulated station D > Q2a, b away from the station under test and such that a transmission from C to D is CCI protected.
	9	send	RF	UNI_BURST_a (sdf:= 1; ro:= 19; lg:= 0; pr:= 0; s:= add_C; d:= add_D) in slot beginning at time = sync_time + 60 - 18 x 60/M1	Ua	Send a unicast burst from station C > Q2a, b away from A, reserving a slot for transmission to station D. The distance from the station under test (station A) to station D is > (CCI ratio) times the distance from station B to station D, so that the transmission from B to D is CCI protected.
	10	await		time:= sync_time + 60		
	11	macro		M_RAND_ACC_SL (slots:= 4) at time = sync_time + 60		Queue a random access transmission over 4 slots.
	12	verify	RF	No RAND_ACC_DATA_a (s = add_A) transmitted in slot beginning at time = sync_time + 60	Ra	Verify that a random access transmission is not made in this slot.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	13	verify	RF	RAND_ACC_DATA_a (s = add_A) transmitted in slot beginning at time = sync_time + 60 + 1 x 60/M1 **AND** time = sync_time + 60 + 2 x 60/M1 **AND** time = sync_time + 60 + 3 x 60/M1	Ra	Verify that random access transmissions are made in these slots.
postamble	14	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	15	send	VSS	SET PARAMETERS (TM2:= 1 500; p:= 64/256)		Reset to default values.
Comments:						

Test Case Name:	**Null_Reservation**					
Purpose:	**To demonstrate that no slot is reserved following the receipt of a null reservation.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_b (pt:= 1; po:= 0; s:= add_B)	Sb	Send a sync burst (burst length 1) from a simulated station B, reserving the same transmission slot in the next superframe, but thereafter terminating the stream.
	5	record	RF	periodic_start:= time at beginning of slot containing the sync burst		Provides a reference time for the next burst from station B.
	6	macro		M_RAND_ACC (sf:= 5)		Queue random access transmissions over 5 superframes.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	
	8	await		time = periodic_start + 60		Wait for the start of the next superframe.
	9	send	RF	NULL_RES_a (s = add_B) in slot beginning at time = periodic_start + 60	Na	Send a null reservation from station B (burst length 1).
	10	rep 4xM1		n:= 1		Repeat over 4 superframes.
	11	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = periodic_start + (n + M1) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots.
	12	endrep		n:= n + 1		
postamble	13	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	14	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:				Periodic_InitialRes		
Purpose:				**To demonstrate that in the absence of any conflicting reservation, a station will maintain a periodic reservation in a constant position in the superframe, with pt = 3 and po = 0, until announcing a further dither.**		
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{min}$:= 8; V11:= 1; V12:= (2/M1) x V11)	Sb	Set up a periodic stream of one-slot messages from the station under test. $TV11_{max}$ equals 8 by default. $TV11_{min}$ set to 8 to cause dither after 8 superframes. V11 set to 1. V12 set to minimum; equals dither range of ±1.
	4	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	5	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_b (s = add_A)	Sb	Define a reference time to measure relative times from during the test.
	6	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + 7 x 60	Sb	For the sync burst before the first dither, record the po value in order to know where the stream will be in the following superframe.
	7	record	RF	PO(0):= po of SYNC_BURST_b	Sb	
	8	rep 10		n:= 1		Repeat test n times.
	9	Repx		k:= 1		
	10	verify verify	RF RF	SYNC_BURST_b (s = add_A) is present in slot beginning at time = sync_time + (n x 8 + k - 1 + PO(n - 1)/M1) x 60 pt = 3 and po = 0 in SYNC_BURST_b	Sb	After each dither, verify that the stream continues in the same position in the superframe with pt = 3 and po = 0, until the next dither is announced.
	11	Until		k:= 5; k:= k + 1		
	12	Await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + (n x 8 + 7 + PO(n - 1)/M1) x 60	Sb	For the sync burst before each dither, record the po value in order to know where the stream will be in the following superframe.
	13	Verify		pt = 0		
	14	Record	RF	PO(n):= po of SYNC_BURST_b	Sb	
	15	endrep		n:= n + 1		
postamble	16	Send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	17	Send	VSS	SET PARAMETERS ($TV11_{min}$:= 4; V11:= 1; V12:= 0,1)		Reset to default values.
	18	Send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_NonDitherRes					
Purpose:	**To demonstrate that a station receiving a periodic broadcast reservation specifying no dither will reserve the appropriate slots.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; s:= add_B)	Sb	Send a sync burst (burst length 1) from a simulated station B, reserving the same transmission slot in the next 4 superframes.
	5	record	RF	periodic_start:= time at beginning of slot containing the sync burst		Provides a reference time for the next burst from station B.
	6	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 5 superframes.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	
	8	await		time = periodic_start + 60		Wait for the start of the next superframe.
	9	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; s:= add_B) in slot beginning at time = periodic_start + 60	Sb	Send a sync burst (burst length 1) from station B in the reserved slot reserving the same transmission slot in the next 4 superframes.
	10	rep 4 x M1		n:= 1		Repeat over 4 superframes.
	11	 verify verify	 RF RF	**IF** n = {M1, 2 x M1, 3 x M1, 4 x M1} **THEN** no transmission present in slot beginning at time = periodic_start + (n + M1) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = add_A) in slot beginning at time = periodic_start + (n + M1) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the reserved slots in the four superframes following the sync burst.
	12	endrep		n:= n + 1		
postamble	13	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	14	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:				Periodic_Potential		
Purpose:	**To demonstrate that a station's reservation table will also record potential reservations, defined as the M1, 2 M1, 3 M1 and 4 M1 slots after a slot, for which no transmission has been decoded.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; s:= add_B)	Sb	Send a sync burst (burst length 1) from a simulated station B, reserving the same transmission slot in the next 4 superframes.
	5	record	RF	periodic_start:= time at beginning of slot containing the sync burst		Provides a reference time for the next burst from station B.
	6	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 5 superframes.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	
	8	await		time = periodic_start + 60		Wait for the start of the next superframe.
	9	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; s:= add_B) AND WITH SAME RF SIGNAL POWER SYNC_BURST_b (pt:= 3; po:= 0; s:= add_C) in slot beginning at time = periodic_start + 60	Sb	Send a sync burst (burst length 1) from station B, and a sync burst (burst length 1) from station C with the same RF signal power, in the reserved slot reserving the same transmission slot in the next 4 superframes. This should result in no burst being decoded in this slot, but should result in potential reservations being made.
	10	rep 4 x M1		n:= 1		Repeat over 4 superframes.
	11	 verify verify	 RF RF	**IF** n = {M1, 2 x M1, 3 x M1, 4 x M1} **THEN** no transmission present in slot beginning at time = periodic_start + (n + M1) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = add_A) in slot beginning at time = periodic_start + (n + M1) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the slots with potential reservations in the four superframes following the un-decoded slot.
	12	endrep		n:= n + 1		
postamble	13	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	14	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_DitherRes					
Purpose:	**To demonstrate that a station receiving a periodic broadcast reservation specifying dither will reserve the appropriate slots.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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)		100 % chance of transmission on access.
test body	4	rep 3		m:= 0; l():= {1; 0; 0} k():= {1; 1; 0} PO():= {50; -100; 25}		Set up loop to repeat test for different values of the periodic timer and the periodic offset Vectors set up to point to relevant dithered slots (with respect to pt) in the verify statement.
	5	send	RF	SYNC_BURST_b (pt:= m; po:= PO; s:= add_B)	Sb	Send a sync burst (burst length 1) from a simulated station B specifying dither in the m + 1th superframe following the current superframe.
	6	record	RF	periodic_start:= time at beginning of slot containing the sync burst		Provides a reference time for the reserved slots of station B.
	7	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 5 superframes.
	8	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	9	await		time = periodic_start + 60		Wait until 60 s after the sync burst from station B.
	10	rep 4 x M1		n:= 1		Verify over 4 superframes.
	11	 verify verify	 RF RF	**IF** n = {PO x l, M1 + (PO x k), (2 x M1) + PO, (3 x M1) + PO} **THEN** no transmission present in slot beginning at time = periodic_start + (n + M1) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = add_A) in slot beginning at time = periodic_start + (n + M1) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the reserved slots (i.e. original reserved slots and dithered slots).
	12	endrep		n:= n + 1		Repeat verification for next slot loop.
	13	wait		60 s		Wait until all the random access transmissions have cleared.
	14	endrep		m:= m + 1		Repeat test with new values loop.
postamble	15	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	16	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_DitherRange					
Purpose:	**To demonstrate that a station will maintain a periodic stream within the dither range in accordance with the V11 and V12 parameters.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{min}$:= 1; $TV11_{max}$:= 1; V11:= 1; V12:= (2/M1) x V11)	Sb	Set up a periodic stream of one-slot messages from the station under test. TV11 reservation hold timer set to cause dither after every superframe. V11 set to 1. V12 set to minimum; equals dither range of ±1.
	4	rep 10		n:= 1		Repeat test 10 times to generate statistical sample.
	5	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	6	record	RF	sync_time(n):= time at beginning of slot of n^{th} SYNC_BURST_b (s = add_A) diff_time:= sync_time(n) - sync_time(1) - (n - 1) x 60 slot_diff(n):= diff_time x M1/60	Sb	Record the time of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each ct_slot and the ct_slot of the first burst and transpose to a common time frame. Convert time differences to slot differences.
	7	endrep		n:= n + 1		
	8	verify		MAX(slot_diff(n)) - MIN(slot_diff(n)) ≤ V12 x M1/V11		Verify (RF) that the transmission is always made within the specified dither range.
	9	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	10	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{min}$:= 1; $TV11_{max}$:= 1; V11:= 1; V12:= (4/M1) x V11)	Sb	Set up a periodic stream of one-slot messages from the station under test. TV11 reservation hold timer set to cause dither after every superframe. V11 set to 1. V12 increased for repeat of above test; equals dither range of ±2.
	11	rep 10		n:= 1		Repeat test 10 times to generate statistical sample.
	12	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	13	record	RF	sync_time(n):= time at beginning of slot of n^{th} SYNC_BURST_b (s = add_A) diff_time:= sync_time(n) - sync_time(1) - (n - 1) x 60 slot_diff(n):= diff_time x M1/60	Sb	Record the time of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each ct_slot and the ct_slot of the first burst and transpose to a common time frame. Convert time differences to slot differences.
	14	endrep		n:= n + 1		
	15	verify		MAX(slot_diff(n)) - MIN(slot_diff(n)) ≤ V12 x M1/V11		Verify (RF) that the transmission is always made within the specified dither range.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
postamble	16	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	17	send	VSS	SET PARAMETERS ($TV11_{min}$:= 4; $TV11_{max}$:= 8; V11:= 1; V12:= 0,1)		Reset to default values.
	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_DitherOffset_A					
Purpose:	**To demonstrate that in the absence of a conflicting reservation, a station will announce a dither to a periodic stream three superframes before the dither occurs.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{min}$:= 8; V11:= 1; V12:= (2/M1) x V11)	Sb	Set up a periodic stream of one-slot messages from the station under test. $TV11_{max}$ equals 8 by default. $TV11_{min}$ set to 8 to cause dither after 8 superframes. V11 set to 1. V12 set to minimum; equals dither range of ±1.
	4	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	5	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_b (s = add_A)	Sb	Define a reference time to measure relative times from during the test.
	6	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + 7 x 60	Sb	For the sync burst before the first dither, record the po value in order to know where the stream will be in the following superframe.
	7	record	RF	PO(0):= po of SYNC_BURST_b	Sb	
	8	rep 3		n:= 1		Repeat test n times.
	9	repx		k:= 1		
	10	verify	RF	SYNC_BURST_b (s = add_A) is present in slot beginning at time = sync_time + (n x 8 + k - 1 + PO(n - 1)/M1) x 60	Sb	Verify that after a dither is announced, the stream dithers to the announced slot.
	11	until		k:= 5; k:= k + 1		
	12	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + (n x 8 + 5 + PO(n - 1)/M1) x 60	Sb	
	13	verify		pt = 2		Verify that a dither is first announced by a transmission with pt = 2.
	14	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + (n x 8 + 7 + PO(n - 1)/M1) x 60	Sb	For the sync burst before each dither, record the po value in order to know where the stream will be in the following superframe.
	15	verify		pt = 0		
	16	record	RF	PO(n):= po of SYNC_BURST_b	Sb	
	17	endrep		n:= n + 1		
postamble	18	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	19	send	VSS	SET PARAMETERS ($TV11_{min}$:= 4; V11:= 1; V12:= 0,1)		Reset to default values.
	20	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:				Periodic_DitherOffset_B		
Purpose:	**To demonstrate that in the absence of a conflicting reservation, following announcement of a dither to a periodic stream, the same dithered slot will be reserved by each of the subsequent two transmissions, containing decrementing values of pt.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{min}$:= 8; V11:= 1; V12:= (2/M1) x V11)	Sb	Set up a periodic stream of one-slot messages from the station under test. $TV11_{max}$ equals 8 by default. $TV11_{min}$ set to 8 to cause dither after 8 superframes. V11 set to 1. V12 set to minimum; equals dither range of ±1.
	4	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	5	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_b (s = add_A)	Sb	Define a reference time to measure relative times from during the test.
	6	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + 7 x 60	Sb	For the sync burst before the first dither, record the po value in order to know where the stream will be in the following superframe.
	7	record	RF	PO(0):= po of SYNC_BURST_b	Sb	
	8	rep 3		n:= 1		Repeat test n times.
	9	repx		k:= 1		
	10	verify	RF	SYNC_BURST_b (s = add_A) is present in slot beginning at time = sync_time + (n x 8 + k - 1 + PO(n - 1)/M1) x 60	Sb	Verify that after a dither is announced, the stream dithers to the announced slot.
	11	until		k:= 5; k:= k + 1		
	12	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + (n x 8 + 5 + PO(n - 1)/M1) x 60	Sb	
	13	verify	RF	pt = 2		Verify that a dither is first announced by a transmission with pt = 2.
	14	record	RF	PO2(n):= po of SYNC_BURST_b	Sb	Record value of po given when pt = 2.
	15	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + (n x 8 + 6 + PO(n - 1)/M1) x 60	Sb	
	16	verify	RF	pt = 1		
	17	record	RF	PO1(n):= po of SYNC_BURST_b	Sb	Record value of po given when pt = 1.
	18	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + (n x 8 + 7 + PO(n - 1)/M1) x 60	Sb	For the sync burst before each dither, record the po value in order to know where the stream will be in the following superframe.
	19	verify	RF	pt = 0		
	20	record	RF	PO(n):= po of SYNC_BURST_b	Sb	
	21	verify		PO2(n) = PO1(n) = PO(n)		Verify that following announcement of a dither by a transmission with pt = 2, the same value of po is contained in subsequent transmissions with pt = 1 and pt = 0.
	22	endrep		n:= n + 1		
postamble	23	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	24	send	VSS	SET PARAMETERS ($TV11_{min}$:= 4; V11:= 1; V12:= 0,1)		Reset to default values.
	25	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_DitherOffset_C					
Purpose:	**To demonstrate that a station will always dither away from the current transmission slot.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{min}$:= 8; V11:= 1; V12:= (2/M1) x V11)	Sb	Set up a periodic stream of one-slot messages from the station under test. $TV11_{max}$ equals 8 by default. $TV11_{min}$ set to 8 to cause dither after 8 superframes. V11 set to 1. V12 set to minimum; equals dither range of ±1.
	4	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	5	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_b (s = add_A)	Sb	Define a reference time to measure relative times from during the test.
	6	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + 7 x 60	Sb	For the sync burst before the first dither, record the po value in order to know where the stream will be in the following superframe.
	7	record	RF	PO(0):= po of SYNC_BURST_b	Sb	
	8	rep 3		n:= 1		Repeat test n times.
	9	repx		k:= 1		
	10	verify	RF	SYNC_BURST_b (s = add_A) is present in slot beginning at time = sync_time + (n x 8 + k - 1 + PO(n - 1)/M1) x 60	Sb	Verify that after a dither is announced, the stream dithers to the announced slot.
	11	until		k:= 5; k:= k + 1		
	12	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + (n x 8 + 5 + PO(n - 1)/M1) x 60	Sb	
	13	verify	RF	pt = 2		Verify that a dither is first announced by a transmission with pt = 2.
	14	record	RF	PO2(n):= po of SYNC_BURST_b	Sb	Record value of po given when pt = 2.
	15	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + (n x 8 + 6 + PO(n - 1)/M1) x 60	Sb	
	16	verify	RF	pt = 1		
	17	record	RF	PO1(n):= po of SYNC_BURST_b	Sb	Record value of po given when pt = 1.
	18	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + (n x 8 + 7 + PO(n - 1)/M1) x 60	Sb	For the sync burst before each dither, record the po value in order to know where the stream will be in the following superframe.
	19	verify	RF	pt = 0		
	20	record	RF	PO(n):= po of SYNC_BURST_b	Sb	
	21	verify		PO2(n) ≠ 0; PO1(n) ≠ 0; PO(n) ≠ 0		Verify (RF) that when a dither is announced by a transmission with pt = 0, 1 or 2, a non-zero value of po is specified, so that the station will dither away from the current transmission slot.
	22	endrep		n:= n + 1		
postamble	23	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	24	send	VSS	SET PARAMETERS ($TV11_{min}$:= 4; V11:= 1; V12:= 0,1)		Reset to default values.
	25	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Periodic_DitherOffset_D**					
Purpose:	**To demonstrate that following announcement of a dither to a periodic stream, the transmission slot will be adjusted to occupy the reserved slot.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{min}$:= 8; V11:= 1; V12:= (2/M1) x V11)		Set up a periodic stream of one-slot messages from the station under test. $TV11_{max}$ equals 8 by default. $TV11_{min}$ set to 8 to cause dither after 8 superframes. V11 set to 1. V12 set to minimum; equals dither range of ±1.
	4	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	5	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_b (s = add_A)	Sb	Define a reference time to measure relative times from during the test.
	6	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + 7 x 60	Sb	For the sync burst before the first dither, record the po value in order to know where the stream will be in the following superframe.
	7	record	RF	PO(0):= po		
	8	rep 3		n:= 1		Repeat test n times.
	9	repx		k:= 1		
	10	verify	RF	SYNC_BURST_b (s = add_A) is present in slot beginning at time = sync_time + (n x 8 + k - 1 + PO(n - 1)/M1) x 60	Sb	Verify that after a dither is announced by a transmission with pt = 0, 1, or 2, the stream dithers to the announced slot.
	11	until		k:= 5; k:= k + 1		
	12	await	RF	SYNC_BURST_b (s = add_A) beginning at time = sync_time + (n x 8 + 7 + PO(n - 1)/M1) x 60	Sb	For the sync burst before each dither, record the po value in order to know where the stream will be in the following superframe.
	13	verify	RF	pt = 0		
	14	record	RF	PO(n):= po		
	15	endrep		n:= n + 1		
postamble	16	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	17	send	VSS	SET PARAMETERS ($TV11_{min}$:= 4; V11:= 1; V12:= 0,1)		Reset to default values.
	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_IndependentStreams					
Purpose:	**To demonstrate that separate streams of periodic broadcasts dither independently.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{min}$:= 1; $TV11_{max}$:= 1; V11:= 3)	Sb	Set up a series of periodic streams of one-slot messages from the station under test. TV11 reservation hold timer set to cause dither every superframe. V11 set to 3 bursts within M1 slots.
	4	rep 10		n:= 1		Record the times of the sync bursts in each of the three streams for 10 superframes.
	5	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	6	record	RF	s1_time_(n):= time at beginning of slot containing SYNC_BURST_b	Sb	
	7	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	8	record	RF	s2_time_(n):= time at beginning of slot containing SYNC_BURST_b	Sb	
	9	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	10	record	RF	s3_time_(n):= time at beginning of slot containing SYNC_BURST_b	Sb	
	11	record		diff1(n):= s1_time_(n) - s1_time_(n - 1)		
	12	record		diff2(n):= s2_time_(n) - s2_time_(n - 1)		
	13	record		diff3(n):= s3_time_(n) - s3_time_(n - 1)		
	14	endrep		n:= n + 1		
	15	rep 10		n:= 1		Verify that the streams dither independently.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	16	verify verify verify verify verify verify		**{** diff1(n) ≠ diff2(n) **AND** diff1(n) ≠ diff3(n) **AND** diff2(n) ≠ diff3(n) **}** **OR** **{** **{** **IF** diff1(n) = diff2(n) **THEN** diff1(n - 1) ≠ diff2(n - 1) **}** **AND** **{** **IF** diff1(n) = diff3(n) **THEN** diff1(n - 1) ≠ diff3(n - 1) **}** **AND** **{** **IF** diff2(n) = diff3(n) **THEN** diff2(n - 1) ≠ diff3(n - 1) **}** **}**		
	17	endrep		n:= n + 1		
postamble	18	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	19	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	20	send	VSS	SET PARAMETERS ($TV11_{min}$:= 4; $TV11_{max}$:= 8; V11:= 1)		Reset to default values.
Comments:						

Test Case Name:				Periodic_Replacement		
Purpose:				**To demonstrate that a station receiving a periodic broadcast reservation in a slot previously reserved by a periodic broadcast will replace the previous reservations by those carried in the new transmission.**		
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; s:= add_B)	Sb	Send a sync burst (burst length 1) from a simulated station B reserving the same transmission slot in the next 4 superframes.
	5	record	RF	periodic_start:= time at beginning of slot containing the sync burst		Provides a reference time for the reserved slots of station B.
	6	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 5 superframes.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	8	await		time = periodic_start + 60		Wait for the expected reserved slot for station B.
	9	send	RF	SYNC_BURST_b (pt:= 0; po:= -50) in slot beginning at time = periodic_start + 60	Sb	Send a sync burst (burst length 1) specifying dither in the next superframe.
	10	rep 4 x M1		n:= 1		Verify over 4 superframes.
	11	 verify verify	 RF RF	**IF** n = {M1 - 50, M2 - 50, M3 - 50, M4 - 50} **THEN** no transmission present in slot beginning at time = periodic_start + (n + M1) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = add_A) in slot beginning at time = periodic_start + (n + M1) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the reserved dithered slots.
	12	endrep		n:= n + 1		
postamble	13	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	14	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_Quarantine_A					
Purpose:	**To demonstrate that a periodic reservation issued by a ground station will cause the following slots to be regarded as quarantined by a station receiving the response.**					
Reference:	**1.3.6.4.1 a**					
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; VS1:= 2)		Ensure 100 % chance of transmission on access. VS1 slots reserved as quarantined slots after the ground station periodic reservation.
test body	4	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; a/d:= 0; s:= add_G, address indicating source is a ground station)	Sb	Send a sync burst (burst length 1) from a simulated ground station G reserving the same transmission slot, and the following VS1 ground quarantined slots, in the next 4 superframes. (The ground station is assumed to have its own VS1 value also set to 2.)
	5	record	RF	periodic_start:= time at beginning of slot containing the sync burst		Provides a reference time for the reserved slots of the ground station G.
	6	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 5 superframes.
	7	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	8	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s= add_A)		
	9	repx		n:= 1		
	10	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = start_time + n × 60/M1	Ra	Verify random access transmissions are made by the station under test in slots preceding the reserved slot.
	11	until		time = periodic_start + (M1 - 1) × 60/M1 in previous step; n:= n + 1		End loop in slot immediately preceding reserved slot.
	12	rep 4 × M1		n:= 0		Verify over 4 superframes.
	13	 verify verify	 RF RF	IF n = {0, 1, 2, M1, M1+1, M1+2, (2 × M1), (2 × M1) + 1, (2 × M1) + 2, (3 × M1), (3 × M1) + 1, (3 × M1) + 2} THEN no transmission present in slot beginning at time = periodic_start + (n + M1) × 60/M1 ELSE RAND_ACC_DATA_a (s= add_A) in slot beginning at time = periodic_start + (n + M1) × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the reserved slots and the following ground quarantined slots.
	14	endrep		n:= n + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
postamble	15	send	VSS	SET PARAMETERS (p:= 64/256; VS1:= 4)		Reset to default values.
	16	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_Quarantine_B					
Purpose:	**To demonstrate that a periodic broadcast issued by a ground station greater than VS4 away will not cause slots to be quarantined.**					
Reference:	**1.3.6.4.1 a**					
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.
test body	4	rep 4		n:= {30, 20, 0, 100}		Repeat for different values of the VS4 parameter.
	5	send	VSS	SET PARAMETERS (VS4:= n)		
	6	send	RF	SYNC_BURST_k(3) (pt:= 3; po:= 0; a/d:= 0; s:= add_G, address indicating source is a ground station; lat:= CPR_LAT(0); lon:= CPR_LON(E (n + 10) NM)) (position of ground station G is > VS4 away from station under test)	Sk(3)	Send a sync burst 3 slots in length from a simulated ground station G > VS4 away from the station under test, reporting the ground station's position.
	7	record	RF	periodic_start:= time at beginning of slot containing the sync burst		Provides a reference time for the reserved slots of the ground station G.
	8	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 4 superframes.
	9	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	10	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s= add_A)	Ra	
	11	repx		n:= 1		
	12	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = start_time + n × 60/M1	Ra	Verify random access transmissions are made by the station under test in slots preceding the reserved slot.
	13	until		time = periodic_start + (M1 - 1) × 60/M1 in previous step; n:= n + 1		End loop in slot immediately preceding reserved slot.
	14	rep 4 × M1		p:= 0		
	15	 verify verify	 RF RF	IF p = {0, 1, 2} THEN no transmission present in slot beginning at time = periodic_start + (p + M1) × 60/M1 ELSE RAND_ACC_DATA_a (s= add_A) in slot beginning at time = periodic_start + (p + M1) × 60/M1	Ra	Verify that random access transmissions are not made by the station under test in the slots reserved by the ground station, but are made in the slots immediately following the reserved slots.
	16	endrep		p:= p + 1		
	17	endrep		next n		
postamble	18	send	VSS	SET PARAMETERS (p:= 64/256; VS4:= 300 NM)		Reset to default values.
	19	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Periodic_Availability_A**					
Purpose:	**To demonstrate that a station will take account of the availability of the current transmission slot when dithering to a new slot.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b (V12:= (10/M1) x V11)	Sb	Set up a periodic stream of one-slot messages from the station under test. V12 set to give dither range of ±5.
	4	await	RF	SYNC_BURST_b (s = add_A; pt = 2; po ≠ 0)	Sb	
	5	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_b (s = add_A; pt = 2; po ≠ 0) PO:= po	Sb	Define a reference time to measure relative times from during the test. Record value of po indicating where the station will dither to.
	6	await		time = sync_time + (2 + 10/M1) x 60		
	7	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; s:= add_B; cprf:= 0; lat:= CPR_LAT(0); lon:= CPR_LON(E 365 NM)) in slot beginning at time = sync_time + (2 + 10/M1) x 60	Sa	Send a sync burst from a simulated station B > 360 NM away from the station under test.
	8	await		time = sync_time + (2 + 20/M1) x 60		
	9	send	RF	SYNC_BURST_a (pt:= 2; po:= PO - 20; s:= add_B; cprf:= 1; lat:= CPR_LAT(0); lon:= CPR_LON(E 365 NM)) in slot beginning at time = sync_time + (2 + 20/M1) x 60	Sa	Send a sync burst from a simulated station B > 360 NM away from the station under test. The burst specifies dither to the same slot that the station under test has announced it will dither to, but two superframes later.
	10	await		time:= sync_time + (3 + PO/M1) x 60		
	11	verify	RF	SYNC_BURST_b (s:= add_A) present in slot beginning at time:= sync_time + (3 + PO/M1) x 60	Sb	Verify that the stream from the station under test has dithered into the specified slot.
	12	 verify verify	 RF Rf	For SYNC_BURST_b (s:= add_A) pt = 1 **AND** po ≠ 0	Sb	Verify that the sync burst from the station under test will dither after the following superframe so as to avoid the slot reserved by station B in two superframe's time.
postamble	13	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	14	send	VSS	SET PARAMETERS (V12:= 0,1)		Reset to default values.
	15	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Periodic_Availability_B**					
Purpose:	**To demonstrate that when the current transmission slot is occupied at the dither of a periodic broadcast, the slot availability is determined from the first occupancy of the slot by a different station.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{max}$:= 4; V11:= 1; V12:= (10/M1) x V11)	Sb	Set up a periodic stream of one-slot messages from the station under test. $TV11_{min}$ equals 4 by default. $TV11_{max}$ set to cause dither after every 4th superframe. V11 set to 1. V12 set to small range; equals dither range of ±5.
	4	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	5	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_b (s = add_A)	Sb	Define a reference time to measure relative times from during the test.
	6	await	RF	time = sync_time + 60		
	7	verify	RF	pt = 2 and po ≠ 0 in SYNC_BURST_b (s = add_A) in slot beginning at time = sync_time + 60	Sb	Verify that the periodic stream is announcing a dither to occur after three superframes.
	8	record	RF	PO:= po in SYNC_BURST_b (s = add_A)	Sb	
	9	await		time = sync_time + 60 + 10 + PO*60/M1		
	10	send	RF	SYNC_BURST_a (pt:= 1; po:= -10; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON (E 365 NM)) in slot beginning at time = sync_time + 60 + 10 + PO*60/M1	Sa	Send a sync burst from a simulated station B, > 360 NM away from the station under test, with pt = 1, which is set to dither into the slot which the station under test has specified but to do so one superframe before the station under test dithers.
	11	await		time = sync_time + 3 x 60 + 20 + PO*60/M1		
	12	send	RF	SYNC_BURST_a (pt:= 2; po:= -20; a/d:= 0; s:= add_C; lat:= CPR_LAT(0); lon:= CPR_LON (E 370 NM)) in slot beginning at time = sync_time + 3 x 60 + 20 + PO*60/M1	Sa	Send a sync burst from a simulated station C, > 360 NM away from the station under test, with pt = 2, which is set to dither into the slot which the station under test has specified but to do so two superframes after the station under test dithers.
	13	await		time = sync_time + 4 x 60 + PO*60/M1		
	14	verify	RF	pt = 1 in SYNC_BURST_b (s = add_A) in slot beginning at time = sync_time + 4 x 60 + PO*60/M1	Sb	
postamble	15	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	16	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	17	send	VSS	SET PARAMETERS ($TV11_{max}$:= 8; V11:= 1; V12:= 0,10)		Reset to default values.
Comments:						

Test Case Name:	**Periodic_Rate**					
Purpose:	**To demonstrate that the station will establish a set of periodic streams at a nominal periodic rate according to the V11 parameter.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	rep 2		k:= {30, 40}		Repeat test for two different values of V11.
	4	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b (V11:= k; V12:= (4/M1) x V11)	Sb	Set up a series of periodic streams of one-slot messages from the station under test. V11 set to k bursts within M1 slots. V12 set to give dither range of ±2.
	5	rep 30		n:= 1		Repeat test k times to generate statistical sample.
	6	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	7	record	RF	sync_time(n):= time at beginning of slot of n^{th} SYNC_BURST_b (s = add_A) diff_time(n):= sync_time(n) - sync_time(1) slot_diff(n):= diff_time(n) x M1/60 - truncate[(n - 1) x (M1/k)]	Sb	Record the time of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each slot and the slot of the first burst in the sequence. Convert time differences to slot differences and subtract the expected n_slot difference, ignoring the unknown n_slot variation of plus or minus 1 slot. Any remaining difference is due to the dither, and to the n_slot variation of plus or minus 1 slot.
	8	endrep		n:= n + 1		
	9	verify		MAX(slot_diff(n)) - MIN(slot_diff(n)) ≤ V12 x M1/V11 + 3		Verify distribution of slots is over the candidate slot range plus 3 slots. The addition of three slots includes (i) 2 slots for the permitted n_slot variation of plus or minus 1 slot; (ii) 1 slot for a rounding difference which may arise if the station under test is beginning its sequence of sync bursts at a different point (this causes a possible difference of 1 slot, as a result of rounding, between the expected and the actual number of slots between successive n_slots).
	10	await		time:= sync_time(1) + 60		Wait for 1 superframe.
	11	rep M1		n:= 0		Repeat for each slot in the next superframe.
	12	verify		**IF** n:= {0, (sync_time(2) - sync_time(1)) x M1/60, (sync_time(3) - sync_time(1)) x M1/60,........, (sync_time(k) - sync_time(1)) x M1/60) **THEN** SYNC_BURST_b (s = add_A) present in slot beginning at time:= sync_time(1) + 60 + n x 60/M1 **ELSE** no transmission in slot	Sb	Verify that the same sync bursts are present in the following superframe.
	13	endrep		n:= n + 1		End loop
	14	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	15	endrep		next k		Repeat for next value of V11.
	16	send	VSS	SET PARAMETERS (V11:= 1; V12:= 0,1)		Reset to default values.
	17	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
Test Case Name:				**Periodic_TV11**		
Purpose:				**To demonstrate that in the absence of any conflicting reservation a station will set the value of TV11 uniformly between the minimum and maximum values.**		
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b ($TV11_{max}$:= 7; V11:= 60)	Sb	Set up a series of periodic streams of one-slot messages from the station under test. $TV11_{min}$ equals 4 by default $TV11_{max}$ set to give four possible values for TV11: 4, 5, 6, and 7. V11 set to give 60 streams.
	4	await	RF	SYNC_BURST_b (s = add_A)	Sb	
	5	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_b (s = add_A)	Sb	Define a reference time to measure relative times from during the test.
	6	repx		n:= 1		
	7	repx		k:= 1; dithered(k):= 0; num(n):= 0		
	8	await	RF	time = sync_time + (n - 1) x 60 + k - 1		
	9	verify	RF	**IF** n < 5 **THEN** SYNC_BURST_b (s = add_A) present in slot	Sb	In the first to fourth superframes check that the slots have not yet dithered.

	10	record	RF	**IF**
				n ≥ 5
				AND
				no transmission present in slot
				AND
				dithered(k) = 0
				THEN
				{
				dithered(k):= 1
				num(n):= num(n) + 1
				}
						In the fifth to eighth superframes, see whether slots have dithered or not. If they have dithered, record in which superframe it happened, and thus count the number of bursts which had each of the four possible TV11 values 4, 5, 6, and 7.

	11	until		k:= 60; k:= k + 1		
	12	until		n:= 8; n:= n + 1		
	13	repx		n:= 1		
	14	await		time = sync_time + (8 x 60 + n - 1)		Wait for ninth superframe.
	15	verify	RF	No transmission present in slot		Confirm that the slots have all dithered from their original positions.
	16	until		n:= 60; n:= n + 1		
	17	repx		n:= 5; chi_squared:= 0		
	18			chi_squared:= chi_squared + $(num(n) - 15)^2/15$		
	19	until		n:= 8; n:= n + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	20	verify		chi_squared < 9.8		Verify that the TV11 values were evenly distributed between $TV11_{min}$ = 4 and $TV11_{max}$ = 7. Value of chi_squared shall be less than 9.8 for confidence that the distribution is uniform (3 degrees of freedom). Thus verify that the time between dithers is set uniformly between $TV11_{min}$ and $TV11_{max}$. The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	21	send	VSS	CANCEL PERIODIC RESERVATION request		Cancel established periodic streams.
	22	send	VSS	SET PARAMETERS ($TV11_{max}$:= 8; V11:= 1)		Reset to default values.
	23	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Periodic_Cancel**					
Purpose:	**To demonstrate that a station receiving a periodic broadcast cancellation in a slot previously reserved for a periodic broadcast will cancel the periodic stream.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; s:= add_B)	Sb	Send a sync burst (burst length 1) from a simulated station B reserving the same transmission slot in the next 4 superframes.
	5	record	RF	periodic_start:= time at beginning of slot containing the sync burst		Provides a reference time for the reserved slots of station B.
	6	macro		M_RAND_ACC (sf:= 5)		Queue random access transmissions over 5 superframes.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	8	await		time = periodic_start + 60		Wait for the expected reserved slot for station B.
	9	send	RF	SYNC_BURST_b (pt:= 0; po:= -128) in slot beginning at time = periodic_start + 60	Sb	Send a sync burst (burst length 1) announcing cancellation of the stream.
	10	rep 4xM1		n:= 1		Verify over 4 superframes.
	11	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = periodic_start + (n + M1) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots over 4 superframes.
	12	endrep		n:= n + 1		
postamble	13	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	14	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_CancelIncremental					
Purpose:	**To demonstrate that upon receipt of an incremental broadcast in a slot expected to contain a periodic broadcast from the same peer station, the periodic stream is cancelled.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; a/d:= 0; s:= add_B)	Sb	Send a sync burst from a simulated station B.
	5	record	RF	sync_time:= time at beginning of slot containing SYNC_BURST_b	Sb	
	6	macro		M_RAND_ACC_SU (sf:= 4)		Queue random access transmissions over 4 superframes.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	8	await		time = sync_time + 60		
	9	send	RF	INCREM_BURST_a (io:= 4; s = add_B) in slot beginning at time = sync_time + 60	Ia	Send an incremental burst from station B in the slot originally reserved for the next sync burst in the periodic stream.
	10	await		time = sync_time + 90		
	11	rep 3xM1		n:= 0		
	12	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = sync_time + n x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots, including those reserved by the block reservation, over 4 superframes.
	13	endrep		n:= n + 1		
postamble	14	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	15	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_CancelUnicast					
Purpose:	**To demonstrate that upon receipt of a unicast request with source/destination flag set to 1 in a slot expected to contain a periodic broadcast from the same peer station, the periodic stream is cancelled.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; a/d:= 0; s:= add_B)	Sb	Send a sync burst from a simulated station B.
	5	record	RF	sync_time:= time at beginning of slot containing SYNC_BURST_b	Sb	
	6	macro		M_RAND_ACC_SU (sf:= 4)		Queue random access transmissions over 4 superframes.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	8	await		time = sync_time + 60		
	9	send	RF	UNI_BURST_a (sdf:= 1; ro:= 5; lg:= 0; pr:= 0; s:= add_B; d:= add_A) in slot beginning at time = sync_time + 60	Ua	Send a unicast burst from station B with source/destination flag set to 1 in the slot originally reserved for the next sync burst in the periodic stream.
	10	await		time = sync_time + 90		
	11	rep 3xM1		n:= 0		
	12	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = sync_time + n x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots, including those reserved by the block reservation, over 4 superframes.
	13	endrep		n:= n + 1		
postamble	14	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	15	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Periodic_CancelQuarantine					
Purpose:	**To demonstrate that a cancellation of a periodic reservation will cause quarantine to be preserved for one superframe following the cancellation.**					
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; VS4:= 140 NM)		Ensure 100 % chance of transmission on access.
test body	4	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; s:= add_G; lat:= CPR_LAT(0); lon:= CPR_LON(E 120 NM)) (position of ground station B is < VS4 away from station under test)	Sa	Send a sync burst from a simulated ground station G < VS4 away from the station under test, reporting G's position.
	5	send	RF	DIR_REQ_a (or:= 0; dt:= 0; nr:= 1; do:= 750; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_B)	Da	Send a directed burst from ground station G, requesting a simulated station B to transmit in the slot do slots after the first slot of the received burst.
	6	record	RF	dir_time:= time at beginning of slot containing DIR_REQ_a	Da	
	7	await		time = dir_time + 10		
	8	send	RF	SYNC_BURST_k(3) (pt:= 3; po:= 0; a/d:= 1; s:= add_B, address indicating source is a mobile; lat:= CPR_LAT(0); lon:= CPR_LON(E 120 NM)) (position of mobile B is < VS4 away from station under test)	Sk(3)	Send a directed sync burst 3 slots in length from a simulated mobile station B with position data showing that it is < VS4 away from the station under test.
	9	record	RF	sync_time:= time at beginning of slot containing SYNC_BURST_k(3)	Sk(3)	
	10	await		time = sync_time + 50		
	11	macro		M_RAND_ACC_SU (sf:= 4)		Queue random access transmissions over 4 superframes.
	12	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	13	await		time = sync_time + 60		
	14	send	RF	SYNC_BURST_k(3) (pt:= 0; po:= 0; a/d:= 1; s:= add_B, address indicating source is a mobile; lat:= CPR_LAT(0); lon:= CPR_LON(E 120 NM)) (position of mobile B is < VS4 away from station under test)	Sk(3)	Send a sync burst from mobile station B cancelling remaining transmissions in the stream.
	15	rep 3 × M1		p:= 0		
	16	 verify verify	 RF RF	**IF** p = {0, 1, 2, 3} **THEN** no transmission present in slot beginning at time = sync_time + 60 + (3 + p) × 60/M1 **ELSE** RAND_ACC_DATA_a (s= add_A) in slot beginning at time = sync_time + 60 + (3 + p) × 60/M1	Ra	Verify that random access transmissions are not made by the station under test in quarantined slots following the periodic cancellation, but are made in all following slots.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	17	endrep		p:= p + 1		
postamble	18	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	19	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Incremental_Reservation_A**					
Purpose:	**To demonstrate that a station receiving an incremental broadcast reservation will reserve the appropriate slots.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	INCREM_BURST_a (io:= 240; s:= add_B)	Ia	Send an incremental burst (burst length 1) from a simulated station B reserving a slot 960 slots away from the t_slot.
	5	record	RF	incremental_start:= time at beginning of slot containing the incremental burst		Provide a reference time for the reserved slot of station B.
	6	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	8	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s = add_A)	Ra	Define a reference time to measure relative times from during the test.
	9	repx		n:= 1		
	10	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = start_time + n x 60/M1	Ra	Verify that random access transmissions are made by the station under test in slots preceding the reserved slot.
	11	until		time = incremental_start + (960 - 1) x 60/M1 in previous step; n:= n + 1		End loop in slot immediately preceding reserved slot (r_slot = t_slot + io x 4).
	12	await		time = incremental_start + 960 x 60/M1		
	13	send	RF	INCREM_BURST_a (io:= 100; s:= add_B) in slot beginning at time = incremental_start + 960 x 60/M1	Ia	Send an incremental burst (bl = 1) from station B in the reserved slot, reserving a slot 400 slots after the t_slot.
	14	repx		n:= 1		
	15	 verify verify	 RF RF	**IF** n = 400 **THEN** no transmission present in slot beginning at time = incremental_start + (n + 960) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = add_A) in slot beginning at time = incremental_start + (n + 960) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the reserved slots.
	16	until		time = start_time + 60; n:= n + 1		Verify until the start of the next superframe after the first random access transmission.
postamble	17	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Incremental_Reservation_B					
Purpose:	**To demonstrate that an incremental broadcast with io= 0 causes no reservation to be made.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 100 NM)) (position of mobile B is < Q2b away from station under test)	Sa	Send a sync burst from a simulated station B with position data showing that it is < Q2b away from the station under test.
	5	send	RF	INCREM_BURST_a (io:= 20; s = add_B)	Ia	Send an incremental burst from station B < Q2b away from the station under test, reserving a slot for B to transmit in.
	6	record	RF	inc_time:= time at beginning of slot containing INCREM_BURST_a	Ia	
	7	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	8	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	9	await		time = inc_time + 80		
	10	send	RF	INCREM_BURST_a (io:= 0; s = add_B)	Ia	Send an incremental burst from station B with io = 0.
	11	rep M1		p:= 0		
	12	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = inc_time + 81 + p x 60/M1	Ra	Verify that random access transmissions are made by the station under test in consecutive slots for 1 superframe.
	13	endrep		p:= p + 1		
postamble	14	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	15	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Incremental_Request					
Purpose:	**To demonstrate that a station will select and reserve a series of future transmission slots by means of the incremental broadcast protocol.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
	3	send	VSS	SET PARAMETERS (V21:= 2; V22:= 12)		V21 (nominal incremental reserved slot position) set to 2 s. V22 (max incremental dither range) set to 12; gives maximum dither range of 150 ± 12 after the incremental broadcast transmission slot (allowed slots of 140, 144, 148, 152, 156, 160).
Test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for an incremental broadcast reservation.
	6	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO(0):= io contained in INCREM_BURST _a (s = add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	record		no_IO(m):= 0 for m:= {140, 144, 148, 152, 156, 160}		Initialize the number of slots in each candidate slot position to zero.
	9	rep 60		n:= 1		
	10	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	11	verify		INCREM_BURST_a (s = add_A) occupies slot beginning at time = current_inc_time + IO(n - 1)	Ia	
	12	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s = add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	13	record verify record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) IO(n) is in the range {140, 144, 148, 152, 156, 160} no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Verify IO(n) is in the expected range. Record the frequency of occurrence of slots in each candidate slot position.
	14	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value in the candidate range. Initialize chi_squared.
	15	record		chi_squared:= chi_squared + (no_IO(m) - 10)^2/10		The distribution is tested for uniformity by calculating the value of chi_squared.
	16	endrep		m:= m + 4		
	17	verify		chi_squared < 13,4		Value of chi_squared shall be less than 13,4 for confidence that the distribution is uniform (5 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
	18	endrep		n:= n + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
postamble	19	send	VSS	SET PARAMETERS (V21:= 1; V22:= 18)		Reset to default values.
	20	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Incremental_SlotSel**					
Purpose:	**To demonstrate that a slot is selected for an incremental broadcast reservation from the appropriate candidate range.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
	3	send	VSS	SET PARAMETERS (V22:= 12)		V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
Test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the incremental broadcast reservation.
	6	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	7	rep 60		n:= 1		Repeat 50 times.
	8	await	RF	INCREM_BURST_a (s = add_A)	Ia	Wait for the next incremental broadcast reservation.
	9	record	RF	IO(n):= io contained in INCREM_BURST _a (s = add_A) no_IO(4 x IO(n)):= no_IO(4 x IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	10	endrep		n:= n + 1		
	11	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	12	record		chi_squared:= chi_squared + $(\text{no\_IO(m)} - 10)^2/10$		The distribution is tested for uniformity by calculating the value of chi_squared.
	13	endrep		m:= m + 4		
	14	verify		chi_squared < 13,4		Value of chi_squared shall be less than 13,4 for confidence that the distribution of the reserved slot over the candidate slots is uniform (5 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	15	send	VSS	SET PARAMETERS (V22:= 18)		Reset to default values.
	16	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Combined_Reservation					
Purpose:	**To demonstrate that receipt of a combined periodic and incremental broadcast reservation causes the appropriate slots to be reserved.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON (E 100 NM)) (position of mobile B is < Q2b away from station under test)	Sa	Send a sync burst from a simulated station B with position data showing that it is < Q2b away from the station under test.
	5	send	RF	INCREM_BURST_a (io:= 20; s = add_B)	Ia	Send an incremental burst from station B < Q2b away from the station under test, reserving a slot for B to transmit in.
	6	record	RF	inc_time:= time at beginning of slot containing INCREM_BURST_a	Ia	
	7	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	8	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	9	await		time = inc_time + 80		
	10	send	RF	INCREM_BURST_a (io:= 0; s = add_B)	Ia	Send an incremental burst from station B with io = 0.
	11	rep M1		p:= 0		
	12	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = inc_time + 81 + p x 60/M1	Ra	Verify that random access transmissions are not made by the station under test in quarantined slots following the periodic cancellation, but are made in all following slots.
	13	endrep		p:= p + 1		
postamble	14	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	15	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**BND_Reservation**					
Purpose:	**To demonstrate that reception of a BND reservation causes the appropriate slots to be reserved.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 100 NM)) (position of mobile B is < Q2b away from station under test)	Sa	Send a sync burst from a simulated station B with position data showing that it is < Q2b away from the station under test.
	5	send	RF	BND_DELAYED_a (nd:= 5)	BDa	Send a delayed burst from station B containing a BND reservation.
	6	record	RF	bnd_time1:= time at beginning of slot containing BND_DELAYED_a	BDa	
	7	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	8	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	9	await		time = bnd_time1 + M1 - 128 - 20		
	10	verify	RF	No transmission by station under test in slot beginning at time = bnd_time1 + M1 - 128 - 20		Verify that no transmission is made by the station under test in the slot reserved by the BND reservation.
	11	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 100 NM)) (position of mobile B is < Q2b away from station under test)	Sa	Send a sync burst from a simulated station B with position data showing that it is < Q2b away from the station under test.
	12	send	RF	BND_LONG_b (nd:= 20)	BDb	Send a single slot burst from station B containing a BND reservation.
	13	record	RF	bnd_time2:= time at beginning of slot containing BND_LONG_b	BDb	
	14	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	15	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	16	await		time = bnd_time2 + M1 - 128 - 80		
	17	verify	RF	No transmission by station under test in slot beginning at time = bnd_time2 + M1 - 128 - 80		Verify that no transmission is made by the station under test in the slot reserved by the BND reservation.
postamble	18	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	19	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Unicast_Reservation_A**					
Purpose:	**To demonstrate that reception of a point-to-point unicast reservation for the destination station to transmit causes the appropriate slots to be reserved.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 200 NM)) (position of mobile B is > Q2a away from station under test)	Sa	Send a sync burst from a simulated station B with position data showing that it is > Q2a away from the station under test.
	5	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 0; s:= add_C; lat:= CPR_LAT(0); lon:= CPR_LON(E 100 NM)) (position of mobile C is < Q2a away from station under test)	Sa	Send a sync burst from a simulated station C with position data showing that it is < Q2a away from the station under test.
	6	send	RF	UNI_BURST_a (sdf:= 0; ro:= 100; lg:= 0; pr:= 0; s:= add_B; d:= add_C)	Ua	Send a unicast burst from station B to station C, with sdf = 0, reserving a slot for C to transmit.
	7	record	RF	uni_time:= time at beginning of slot containing UNI_BURST_a	Ua	
	8	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	9	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	10	verify	RF	No transmission by station under test in slot beginning at time = uni_time + 101		Verify that no transmission is made by the station under test in the slot reserved by the unicast reservation.
postamble	11	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	12	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Unicast_Reservation_B					
Purpose:	**To demonstrate that a reception of a point-to-point unicast reservation for the source station to transmit causes the appropriate slots to be reserved.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 100 NM)) (position of mobile B is < Q2a away from station under test)	Sa	Send a sync burst from a simulated station B with position data showing that it is < Q2a away from the station under test.
	5	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 0; s:= add_C; lat:= CPR_LAT(0); lon:= CPR_LON(E 200 NM)) (position of mobile C is > Q2a away from station under test)	Sa	Send a sync burst from a simulated station C with position data showing that it is > Q2a away from the station under test.
	6	send	RF	UNI_BURST_a (sdf:= 1; ro:= 100; lg:= 0; pr:= 0; s:= add_B; d:= add_C)	Ua	Send a unicast burst from station B to station C, with sdf = 1, reserving a slot for B to transmit.
	7	record	RF	uni_time:= time at beginning of slot containing UNI_BURST_a	Ua	
	8	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	9	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	10	verify	RF	No transmission by station under test in slot beginning at time = uni_time + 101		Verify that no transmission is made by the station under test in the slot reserved by the unicast reservation.
postamble	11	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	12	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Unicast_Reservation_C**					
Purpose:	**To demonstrate that a reception of a broadcast unicast reservation causes the appropriate slots to be reserved.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 100 NM)) (position of mobile B is < Q2b away from station under test)	Sa	Send a sync burst from a simulated station B with position data showing that it is < Q2b away from the station under test.
	5	send	RF	UNI_BURST_c (ro:= 100; lg:= 0; pr:= 0; s:= add_B)	Uc	Send a unicast burst from station B to a broadcast address, reserving a slot for B to broadcast.
	6	record	RF	uni_time:= time at beginning of slot containing UNI_BURST_c	Uc	
	7	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	8	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	9	verify	RF	No transmission by station under test in slot beginning at time = uni_time + 101		Verify that no transmission is made by the station under test in the slot reserved by the unicast reservation.
postamble	10	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	11	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Unicast_Reservation_D**					
Purpose:	**To demonstrate that a station applying the slot selection criteria will exclude any slot reserved by another station using the unicast request protocol with sdf = 1.**					
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.
test body	4	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 190 NM)) (position of mobile B is > Q2a away from station under test)	Sa	Send a sync burst from a simulated station B with position data showing that it is > Q2a away from the station under test.
	5	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 0; s:= add_C; lat:= CPR_LAT(0); lon:= CPR_LON(E 200 NM)) (position of mobile C is > Q2a away from station under test)	Sa	Send a sync burst from a simulated station C with position data showing that it is > Q2a away from the station under test. The position of station C is such that a point-to-point transmission from station B to C is CCI protected.
	6	send	RF	UNI_BURST_a (sdf:= 1; ro:= 100; lg:= 0; pr:= 0; s:= add_B; d:= add_C)	Ua	Send a unicast burst from station B to station C, with sdf = 1, reserving a slot for B to transmit.
	7	record	RF	uni_time:= time at beginning of slot containing UNI_BURST_a	Ua	
	8	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	9	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	10	verify	RF	No transmission by station under test in slot beginning at time = uni_time + 101		Verify that no transmission is made by the station under test in the slot reserved by the unicast reservation.
postamble	11	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	12	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Info_Reservation					
Purpose:	**To demonstrate that a station receiving a burst containing an information transfer request reservation addressed to another station will reserve the slots identified for the information transfer and acknowledgement.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	INF_TRANS_a (ro:= 2 000; lg:= 5; ao:= 75; f:= 0; s:= add_B; d:= add_D)	IFa	Send an information transfer burst (burst length 1) from a simulated station B, addressed to a simulated station D. The burst reserves a slot 2 001 slots away from the t_slot for station D to transmit in, and a slot 2 001 + 6 + 75 slots after t_slot for station B to make an acknowledgement to station D.
	5	record	RF	transfer_start:= time at beginning of slot containing the incremental burst		Provides a reference time for the reserved slots.
	6	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	8	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s = add_A)	Ra	Define a reference time to measure relative times from during the test.
	9	repx		n:= 1		
	10	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = start_time + n x 60/M1	Ra	Verify that random access transmissions are made by the station under test in slots preceding the reserved slots.
	11	until		time = transfer_start + 2 000 x 60/M1 in previous step; n:= n + 1		End loop in slot immediately preceding reserved slot (r_slot = t_slot + ro + 1).
	12	rep 81		n:= 0		Verify up to the slot preceding the acknowledgement slot.
	13	 verify verify	 RF RF	**IF** n = {0, 1, 2, 3, 4, 5} **THEN** no transmission present in slot beginning at time = transfer_start + (n + 2 001) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = add_A) in slot beginning at time = transfer_start + (n + 2 001) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the block of reserved slots (6).
	14	endrep		n:= n + 1		
	15	await		time = transfer_start + 2 082 x 60/M1		
	16	send	RF	INF_TRANS_a (ro:= 300; lg:= 10; ao:= 50; f:= 0; d:= address of a station other than the station under test) in slot beginning at time = transfer_start + 2 082 x 60/M1	IFa	Send an information transfer burst (bl = 1) in the acknowledgement slot from station B, addressed to station D, reserving a slot 301 slots after the t_slot for station D to transmit in.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	17	repx		n:= 0		
	18	 verify verify	 RF RF	**IF** n = {301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 362} **THEN** no transmission present in slot beginning at time = transfer_start + (n + 2 082) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = add_A) in slot beginning at time = transfer_start + (n + 2 001) x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the block of reserved slots and the acknowledgement slot.
	19	until		time = start_time + 60; n:= n + 1		Verify until start of the next superframe after the first random access transmission.
postamble	20	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	21	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Autotune_Autonomous_A					
Purpose:	**To demonstrate that an autotune request for autonomous broadcasts on the current frequency with trmt=0 causes a station to send autonomous sync bursts as directed, cancelling the current autonomous transmissions, and also that the default sync burst reporting rate is restored following expiry of the request.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS (V12:= (10/M1) × V11; $TV11_{min}$:= 8)		Set dither range to a minimum (±5) to limit choice of slots. Set TV11 timer to force dither after 8 superframes.
test body	3	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC1.
	4	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC2.
	5	send	RF (GSC2)	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 0; lg:= 0; rcvr:= 0; f:= frequency of GSC1; trmt:= 0; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst on GSC2 from a simulated ground station G and with the transmit control (trmt) flag set to 0, requesting the station under test to autonomously transmit at a rate of 4 bursts per superframe for 5 superframes on GSC1.
	6	record	RF (GSC2)	directed_time:= time at beginning of slot containing DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	7	rep 5		p:= 0		Record the report rate over 5 × M1 slots starting from the slot containing the reservation.
	8	rep M1		n:= 1; m:= 0		Inner loop for recording report rate over 1 superframe.
	9	do	RF (GSC1)	**IF** slot beginning at time:= directed_time + p × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) **THEN** m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on GSC1.
	10	verify	RF (GSC2)	slot beginning at time:= directed_time + p × 60 + n × 60/M1 does not contain SYNC_BURST_c (s= add_A)	Sc	Verify that the station under test cancels default sync burst transmissions on the GSC channel which carried the autotune request (GSC2).
	11	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	12	record		update_rate(p):= m		Record the update rate for each superframe.
	13	endrep		p:= p + 1		Outer loop advancing to the next superframe.
	14	verify		update_rate(p) = 4 for all p		Verify that the station under test transmits at the directed rate on GSC1 for 5 superframes.
	15	rep 2		j:= 0		Record the report rate on GSC1 for a further 2 superframes.
	16	rep M1		n:= 1; m:= 0		Inner loop for recording report rate over 1 superframe.
	17	do	RF (GSC1)	**IF** slot beginning at time = directed_time + (5 + j) × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) **THEN** m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on GSC1.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	18	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	19	record		default_rate(j):= m		Record the report rate for each superframe.
	20	endrep		j:= j + 1		Outer loop advancing to the next superframe.
	21	verify		default_rate(j) = 6 for all j		Verify that following expiry of the autotune reservation, the station under test transmits on GSC1 at the default rate.
postamble	22	send	VSS	SET PARAMETERS (V12:= 0.1; $TV11_{min}$:= 4)		Reset to default value.
Comments:						

Test Case Name:	**Autotune_Autonomous_B**					
Purpose:	**To demonstrate that an autotune request for autonomous broadcasts on a different frequency with trmt=0 causes a station to send autonomous sync bursts as directed, cancelling the current autonomous transmissions, and also that the default sync burst reporting rate is restored following expiry of the request.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	1	rep 2		fa:= {f1, f2}		Repeat for two values of frequency f (initially f1).
Preamble	2	do		M_POWER_UP		Prepare the transceiver for testing.
	3	send	VSS	SET PARAMETERS (V12:= (10/M1) × V11; $TV11_{min}$:= 8)		Set dither range to a minimum (±5) to limit choice of slots. Set TV11 timer to force dither after 8 superframes.
test body	4	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC1.
	5	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC2.
	6	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 0; lg:= 0; rcvr:= 0; f:= fa (not a GSC channel); trmt:= 0; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst on GSC1 from a simulated ground station G and with the transmit control (trmt) flag set to 0, requesting the station under test to autonomously transmit at a rate of 4 bursts per superframe for 5 superframes on frequency fa.
	7	record	RF (GSC1)	directed_time:= time at beginning of slot containing DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	8	rep 5		p:= 0		Record the report rate on frequency fa over 5 × M1 slots starting from the slot containing the reservation.
	9	rep M1		n:= 1; m:= 0		Inner loop for recording report rate over 1 superframe.
	10	do	RF (fa)	IF slot beginning at time:= directed_time + p × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) THEN m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on frequency fa.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	11	verify	RF (GSC1)	slot beginning at time:= directed_time + p × 60 + n × 60/M1 does not contain SYNC_BURST_c (s= add_A)	Sc	Verify that the station under test cancels default sync burst transmissions on the GSC channel which carried the autotune request (GSC1).
	12	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	13	record		update_rate(p):= m		Record the report rate on frequency fa for each superframe.
	14	endrep		p:= p + 1		Outer loop advancing to the next superframe.
	15	verify		update_rate(p) = 4 for all p		Verify that the station under test transmits at the directed rate on frequency fa for 5 superframes.
	16	rep 2		j:= 0		Record the report rate on GSC1 for a further 2 superframes.
	17	rep M1		n:= 1; m:= 0		Inner loop for recording report rate over 1 superframe.
	18	do	RF (GSC1)	IF slot beginning at time = directed_time + (5 + j) × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) THEN m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on GSC1.
	19	verify	RF (fa)	slot beginning at time:= directed_time + p × 60 + n × 60/M1 does not contain SYNC_BURST_c (s= add_A)	Sc	Verify that sync burst transmissions on frequency fa have ceased.
	20	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	21	record		default_rate(j):= m		Record the report rate on GSC1 for each superframe.
	22	endrep		j:= j + 1		Outer loop advancing to the next superframe.
	23	verify		default_rate(j) = 6 for all j		Verify that following expiry of the autotune reservation, the station under test transmits at the default rate on GSC1.
Postamble	24	send	VSS	SET PARAMETERS (V12:= 0.1; $TV11_{min}$:= 4)		Reset to default value.
	25	do		switch off VDL4 transceiver		Switch off transceiver in preparation for second test.
	26	endrep		next fa		Repeat for second frequency.
Comments:						

Test Case Name:	Autotune_Autonomous_C					
Purpose:	**To demonstrate that an autotune request for autonomous broadcasts with trmt=1 causes a station to continue current autonomous transmissions during and after expiry of the request.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS (V12:= (10/M1) × V11; $TV11_{min}$:= 8)		Set dither range to a minimum (±5) to limit choice of slots. Set TV11 timer to force dither after 8 superframes.
test body	3	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC1.
	4	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC2.
	5	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 0; lg:= 0; rcvr:= 0; f:= f3 (not a GSC channel); trmt:= 1; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst on GSC1 from a simulated ground station G and with the transmit control (trmt) flag set to 1, requesting the station under test to autonomously transmit at a rate of 4 bursts per superframe for 5 superframes on frequency f3.
	6	record	RF (GSC1)	directed_time:= time at beginning of slot containing DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	7	rep 5		p:= 0		Record the report rates on frequency f3 and GSC1 over 5 × M1 slots starting from the slot containing the reservation.
	8	rep M1		n:= 1; m:= 0; k:= 0		Inner loop for recording report rate over 1 superframe.
	9	do	RF (f3)	IF slot beginning at time:= directed_time + p × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) THEN m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on frequency f3.
	10	verify	RF (GSC1)	IF slot beginning at time:= directed_time + p × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) THEN k:= k + 1	Sc	Increase the counter k by 1 every time a sync burst is transmitted on GSC1.
	11	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	12	record		update_rate(p):= m		Record the report rate for each superframe for reports on frequency f3.
	13	record		default_rate(p):= k		Record the report rate for each superframe for reports on GSC1.
	14	endrep		p:= p + 1		Outer loop advancing to the next superframe.
	15	verify		update_rate(p) = 4 for all p		Verify that the station under test transmits at the directed rate on frequency f3 for 5 superframes.
	16	verify		default_rate(p) = 6 for all p		Verify that the station under test continues to transmit at the default rate on GSC1.
	17	rep 2		j:= 0		Record the report rate on GSC1 for a further 2 superframes.
	18	rep M1		n:= 1; m:= 0		Inner loop for recording report rate over 1 superframe.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	19	do	RF (GSC1)	IF slot beginning at time = directed_time + (5 + j) × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) THEN m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on GSC1.
	20	verify	RF (f3)	slot beginning at time:= directed_time + p × 60 + n × 60/M1 does not contain SYNC_BURST_c (s= add_A)	Sc	Verify that sync burst transmissions on frequency f3 have ceased.
	21	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	22	record		default_rate_2(j):= m		Record the report rate on GSC1 for each superframe.
	23	endrep		j:= j + 1		Outer loop advancing to the next superframe.
	24	verify		default_rate_2(j) = 6 for all j		Verify that following expiry of the autotune reservation, the station under test transmits at the default rate on GSC1.
Postamble	25	send	VSS	SET PARAMETERS (V12:= 0.1; $TV11_{min}$:= 4)		Reset to default value.
Comments:						

Test Case Name:	**Autotune_Autonomous_D**					
Purpose:	**To demonstrate that an autotune request for autonomous broadcasts on a different frequency with rcvr=00 or 11 causes a station to continue receiving sync bursts on the current frequency.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
Preamble	1	rep 2		rc:= {00, 11}		Repeat for two values of rcvr (initially 00).
	2	do		M_POWER_UP		Prepare the transceiver for testing.
	3	send	VSS	SET PARAMETERS (V12:= (10/M1) × V11; $TV11_{min}$:= 8)		Set dither range to a minimum (±5) to limit choice of slots. Set TV11 timer to force dither after 8 superframes.
test body	4	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC1.
	5	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC2.
	6	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 0; lg:= 0; rcvr:= 0; f:= f4 (not a GSC channel); trmt:= 1; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst on GSC1 from a simulated ground station G and with the transmit control (trmt) flag set to 1, requesting the station under test to autonomously transmit at a rate of 4 bursts per superframe for 5 superframes on frequency f4.
	7	record	RF (GSC1)	directed_time:= time at beginning of slot containing DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	8	rep 2		p:= 0		Record the report rates on frequency f4 and GSC1 over 2 × M1 slots starting from the slot containing the reservation.
	9	rep M1		n:= 1; m:= 0		Inner loop for recording report rate over 1 superframe.
	10	do	RF (f4)	IF slot beginning at time:= directed_time + p × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) THEN m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on frequency f4.
	11	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	12	record		update_rate(p):= m		Record the report rate for each superframe for reports on frequency f4.
	13	endrep		p:= p + 1		Outer loop advancing to the next superframe.
	14	verify		update_rate(p) = 4 for all p		Verify that the station under test transmits at the directed rate on frequency f4 for 2 superframes.
	15	send	RF (GSC1)	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM))	Sa	Send a sync burst on GSC1 from a simulated station B 70 NM away from the station under test.
	16	send	RF (GSC1)	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 80 NM))	Sa	Send a sync burst on GSC1 from a simulated station B 80 NM away from the station under test.
	17	verify	AppOut	LAT DATA OUT = {N 0 NM, N 0 NM} AND LON DATA OUT = {E 70 NM, E 80 NM}		Verify that the station under test receives the sync burst data processes the data and generates the appropriate output for display to the aircrew.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
postamble	18	send	VSS	SET PARAMETERS (V12:= 0.1; $TV11_{min}$:= 4)		Reset to default value.
	19	do		switch off VDL4 transceiver		Switch off transceiver in preparation for second test.
	20	endrep		next rc		Repeat for rcvr:= 11.
Comments:						

Test Case Name:	**Autotune_Autonomous_E**					
Purpose:	**To demonstrate that an autotune request for autonomous broadcasts on a different frequency with rcvr=01 causes a station to receive sync bursts on the indicated frequency.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS (V12:= (10/M1) × V11; $TV11_{min}$:= 8)		Set dither range to a minimum (±5) to limit choice of slots. Set TV11 timer to force dither after 8 superframes.
test body	3	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC1.
	4	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC2.
	5	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 0; lg:= 0; rcvr:= 01; f:= f5 (not a GSC channel); trmt:= 1; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst on GSC1 from a simulated ground station G and with the transmit control (trmt) flag set to 1, requesting the station under test to autonomously transmit at a rate of 4 bursts per superframe for 5 superframes on frequency f5.
	6	record	RF (GSC1)	directed_time:= time at beginning of slot containing DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	7	rep 2		p:= 0		Record the report rates on frequency f5 and GSC1 over 2 × M1 slots starting from the slot containing the reservation.
	8	rep M1		n:= 1; m:= 0		Inner loop for recording report rate over 1 superframe.
	9	do	RF (f5)	IF slot beginning at time:= directed_time + p × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) THEN m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on frequency f5.
	10	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	11	record		update_rate(p):= m		Record the report rate for each superframe for reports on frequency f5.
	12	endrep		p:= p + 1		Outer loop advancing to the next superframe.
	13	verify		update_rate(p) = 4 for all p		Verify that the station under test transmits at the directed rate on frequency f5 for 2 superframes.
	14	send	RF (f5)	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM))	Sa	Send a sync burst on frequency f5 from a simulated station B 70 NM away from the station under test.
	15	send	RF (f5)	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 80 NM))	Sa	Send a sync burst on frequency f5 from a simulated station B 80 NM away from the station under test.
	16	verify	AppOut	LAT DATA OUT = {N 0 NM, N 0 NM} AND LON DATA OUT = {E 70 NM, E 80 NM}		Verify that the station under test receives the sync burst data processes the data and generates the appropriate output for display to the aircrew.
postamble	17	send	VSS	SET PARAMETERS (V12:= 0.1; $TV11_{min}$:= 4)		Reset to default value.
Comments:						

Test Case Name:	Autotune_AutonomousCessation					
Purpose:	**To demonstrate that on cessation of an autotune command directing the reporting rate, normal default behaviour on the GSCs is established.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	send	RF	DIR_REQ_a (or:= 0; dt:= 0; nr:= 60; do:= 0; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_D)	Da	Send a directed rate reservation from a simulated ground station G, requesting a simulated station D to transmit at a rate of 60 bursts per superframe for 1 superframe.
	3	record	RF	dir_time:= time at beginning of slot containing DIR_REQ_a (s= add_G)	Da	Define a reference time for the directed slot burst.
	4	await		time:= dir_time + 10 × 60/M1		
	5	rep 60		n:= 1		
	6	await		SYNC_BURST_c (s= add_A)	Sc	
	7	verify		SYNC_BURST_c (s= add_A) occurs before slot beginning at time:= dir_time + 1 × 60/M1 + n	Sc	
	8	endrep		n:= n + 1		
	9	await	RF	SYNC_BURST_c (s= add_A)	Sc	Await next autonomous sync burst following expiry of the autotune command.
	10	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_c (s= add_A)	Sc	Define a reference time to measure relative times from during the test.
	11	rep 10		n:= 1		Repeat test on GSC1 10 times.
	12	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	
	13	record	RF	IF n = 10 THEN time(n) = time at the beginning of slot containing SYNC_BURST_c (s= add_A)	Sc	
	14	endrep		n:= n + 1		
	15	verify		time(10) - sync_time ≤ 10 × M1		Verify that on GSC1 the station emits at least 1 sync burst per M1 slots.
	16	rep 10		n:= 1		Repeat test on GSC2 10 times.
	17	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	
	18	record	RF	IF n = 10 THEN time(n) = time at the beginning of slot containing SYNC_BURST_c (s= add_A)	Sc	
	19	endrep		n:= n + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	20	verify		time(10) - sync_time ≤ 10 × M1		Verify that on GSC1 the station emits at least 1 sync burst per M1 slots.
postamble	21					Bring test equipment into idle state.
Comments:						

Test Case Name:	**Autotune_Directed_A**					
Purpose:	**To demonstrate that an autotune request directing transmission in specified slots on the current frequency, causes a station to transmit in the directed slots, cancelling current autonomous transitions, and that the default sync burst reporting rate is restored following expiry of the command.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS (V12:= (10/M1) × V11; $TV11_{min}$:= 8)		Set dither range to a minimum (±5) to limit choice of slots. Set TV11 timer to force dither after 8 superframes.
test body	3	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC1.
	4	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC2.
	5	send	RF (GSC2)	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 1125; lg:= 0; rcvr:= 0; f:= frequency of GSC1; trmt:= 0; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst on GSC2 from a simulated ground station G and with the transmit control (trmt) flag set to 0, requesting the station under test to transmit at a rate of 4 bursts per superframe for 5 superframes on GSC1.
	6	record	RF (GSC2)	directed_time:= time at beginning of slot containing DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	7	rep 22500		n:= 1		Verify over the 5 superframes containing the directed reservations.
	8	 verify verify	 RF (GSC1) RF (GSC1)	IF n = {1125, 2250, 3375, 4500, 5625, 6750, 7875, 9000, 10125, 11250, 12375, 13500, 14625, 15750, 16875, 18000, 19125, 20250, 21375, 22500} THEN slot beginning at time = directed_time + n × 60/M1 contains DIR_SYNC_BURST_a (s= add_A) ELSE no transmission in slot beginning at time:= directed_time + n × 60/M1	DSa	Verify that directed sync bursts are transmitted on GSC1 in the reserved slots given by slots do + k × (M1/nr) + j × M1 after the first slot of the received bursts for j = 0 to dt and k = 0 to nr - 1. Verify that there are no transmissions except in the directed slots.
	9	verify	RF (GSC2)	slot beginning at time:= directed_time + p × 60 + n × 60/M1 does not contain SYNC_BURST_c (s= add_A)	Sc	Verify that the station under test cancels default sync burst transmissions on the GSC channel which carried the autotune request (GSC2).
	10	endrep		n:= n + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	11	rep 2		j:= 0		Record the report rate on GSC1 for a further 2 superframes.
	12	rep M1		n:= 1; m:= 0		Inner loop for recording report rate over 1 superframe.
	13	do	RF (GSC1)	IF slot beginning at time = directed_time + (5 + j) × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) THEN m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on GSC1.
	14	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	15	record		default_rate(j):= m		Record the report rate for each superframe.
	16	endrep		j:= j + 1		Outer loop advancing to the next superframe.
	17	verify		default_rate(j) = 1 for all j		Verify that following expiry of the autotune reservation, the station under test transmits on GSC1 at the default rate.
postamble	18	send	VSS	SET PARAMETERS (V12:= 0.1; $TV11_{min}$:= 4)		Reset to default value.
Comments:						

Test Case Name:	**Autotune_Directed_B**					
Purpose:	**To demonstrate that an autotune request directing transmission in specified slots on a different frequency, causes a station to transmit in the directed slots, cancelling current autonomous transitions, and that the default sync burst reporting rate is restored following expiry of the command.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS (V12:= (10/M1) × V11; $TV11_{min}$:= 8)		Set dither range to a minimum (±5) to limit choice of slots. Set TV11 timer to force dither after 8 superframes.
test body	3	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC1.
	4	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC2.
	5	send	RF (GSC2)	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 1125; lg:= 0; rcvr:= 0; f:= f6 (not a GSC channel); trmt:= 0; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst on GSC2 from a simulated ground station G and with the transmit control (trmt) flag set to 0, requesting the station under test to transmit at a rate of 4 bursts per superframe for 5 superframes on frequency f6.
	6	record	RF (GSC2)	directed_time:= time at beginning of slot containing DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	7	rep 22500		n:= 1		Verify over the 5 superframes containing the directed reservations.
	8	 verify verify	 RF (f6) RF (f6)	IF n = {1125, 2250, 3375, 4500, 5625, 6750, 7875, 9000, 10125, 11250, 12375, 13500, 14625, 15750, 16875, 18000, 19125, 20250, 21375, 22500} THEN slot beginning at time = directed_time + n × 60/M1 contains DIR_SYNC_BURST_a (s= add_A) ELSE no transmission in slot beginning at time:= directed_time + n × 60/M1	DSa	Verify that directed sync bursts are transmitted on frequency f6 in the reserved slots given by slots do + k × (M1/nr) + j × M1 after the first slot of the received bursts for j = 0 to dt and k = 0 to nr - 1. Verify that there are no transmissions except in the directed slots.
	9	verify	RF (GSC2)	slot beginning at time:= directed_time + p × 60 + n × 60/M1 does not contain SYNC_BURST_c (s= add_A)	Sc	Verify that the station under test cancels default sync burst transmissions on the GSC channel which carried the autotune request (GSC2).
	10	endrep		n:= n + 1		
	11	rep 2		j:= 0		Record the report rate on GSC1 for a further 2 superframes.
	12	rep M1		n:= 1; m:= 0		Inner loop for recording report rate over 1 superframe.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	13	do	RF (GSC1)	IF slot beginning at time = directed_time + (5 + j) × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) THEN m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on GSC1.
	14	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	15	record		default_rate(j):= m		Record the report rate for each superframe.
	16	endrep		j:= j + 1		Outer loop advancing to the next superframe.
	17	verify		default_rate(j) = 1 for all j		Verify that following expiry of the autotune reservation, the station under test transmits on GSC1 at the default rate.
postamble	18	send	VSS	SET PARAMETERS (V12:= 0.1; $TV11_{min}$:= 4)		Reset to default value.
Comments:						

Test Case Name:	**Autotune_Unsupported**					
Purpose:	**To demonstrate that when an autotune request cannot be supported due to lack of receiver resources, a general failure is issued with the required error code.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS (V12:= (10/M1) × V11; $TV11_{min}$:= 8)		Set dither range to a minimum (±5) to limit choice of slots. Set TV11 timer to force dither after 8 superframes.
test body	3	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC1.
	4	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst on GSC2.
	5	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 0; lg:= 0; rcvr:= 11; f:= f1 (not a GSC channel); trmt:= 1; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst on GSC1 from a simulated ground station G and with the transmit control (trmt) flag set to 0, requesting the station under test to autonomously transmit at a rate of 4 bursts per superframe for 5 superframes on frequency f1.
	6	record	RF (GSC1)	directed_time:= time at beginning of slot containing DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	7	rep 2		p:= 0		Record the report rate on frequency fa over 2 × M1 slots starting from the slot containing the reservation.
	8	rep M1		n:= 1; m:= 0		Inner loop for recording report rate over 1 superframe.
	9	do	RF (f1)	IF slot beginning at time:= directed_time + p × 60 + n × 60/M1 contains SYNC_BURST_c (s= add_A) THEN m:= m + 1	Sc	Increase the counter m by 1 every time a sync burst is transmitted on frequency f1.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	10	endrep		n:= n + 1		Inner loop advancing to the next slot in the superframe.
	11	record		update_rate(p):= m		Record the report rate on frequency fa for each superframe.
	12	endrep		p:= p + 1		Outer loop advancing to the next superframe.
	13	verify		update_rate(p) = 4 for all p		Verify that the station under test transmits at the directed rate on frequency f1 for 2 superframes.
	14	send	RF (GSC2)	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 100; lg:= 0; rcvr:= 11; f:= f2 (not a GSC channel and not equal to f1); trmt:= 1; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A) in slot beginning at time:= sync_time + 60 - 10 × 60/M1	Da	Send a directed burst on GSC1 from a simulated ground station G and with the transmit control (trmt) flag set to 0, requesting the station under test to transmit at a rate of 4 bursts per superframe for 5 superframes on frequency f1.
	15	record	RF (GSC1)	directed_time:= time at beginning of slot containing DIR_REQ_a	Da	Define a reference time to measure relative times from during the test.
	16	verify	RF (f2)	GEN_RESP_a (ok= 0; r-mi= 1010101binary; err= 00 hex; bd= 0; s= add_A; d:= add_G) in slot beginning at time:= directed_time + 100	GRa	Verify that a General Failure burst is sent on frequency f2 in a slot reserved for the station under test by the ground station.
postamble	17	send	VSS	SET PARAMETERS (V12:= 0.1; $TV11_{min}$:= 4)		Reset to default value.
Comments: This test is not applicable if the transceiver implements more than three receiver channels.						

Test Case Name:	Autotune_Reservation					
Purpose:	**To demonstrate that a station receiving a directed request from a ground station addressed to another station will reserve the directed slots.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	DIR_REQ_a (or:= 0; pr_flag:= 0; dt:= 4; nr:= 4; do:= 1 125; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_D)	Da	Send a directed burst from a simulated ground station G, requesting transmission by a simulated station D and specifying slots for D to transmit in.
	5	record	RF	directed_time:= time at beginning of slot containing directed request reservation		Define a reference time to measure relative times from during the test.
	6	macro		M_RAND_ACC_SU (sf:= 6)		Queue random access transmissions over 6 superframes.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	
	8	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s = add_A)	Ra	Define a reference time to measure relative times from during the test.
	9	repx		n:= 1		
	10	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = start_time + n x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots up to the reserved slot.
	11	until		time = directed_time + (do - 1) x 60/M1 in previous step; n:= n + 1		End loop before first directed reservation.
	12	rep 5xm1		n:= 0		Verify over the 5 superframes containing the directed reservations.
	13	 verify verify	 RF RF	**IF** n = {0, 1 125, 2 250, 3 375, 4 500, 5 625, 6 750, 7 875, 9 000, 10 125, 11 250, 12 375, 13 500, 14 625, 15 750, 16 875, 18 000, 19 125, 20 250, 21 375} **THEN** no transmission present in slot beginning at time:= directed_time + (do + n) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = add_A) in slot beginning at time:= directed_time + (do + n) x 60/M1)	Ra	Verify that no transmissions are made in the reserved slots given by slots do + k x (M1/nr) + j x M1 after the first slot of the received burst for j = 0 to dt and k = 0 to nr - 1. Verify that random access transmissions are made by the station under test in all slots except the reserved slots.
	14	endrep		n:= n + 1		
postamble	15	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	16	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Autotune_CancelResp**					
Purpose:	**To demonstrate that a station making a directed request cancellation will provoke the addressed station to cancel the directed broadcast.**					
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.
test body	3	send	RF	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 1125; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_A)	Da	Send a directed burst from a simulated ground station G, requesting the station under test to transmit at a rate of 4 bursts per superframe for 5 superframes in the directed slots, starting in the slot do slots after the first slot of the received burst.
	4	await	RF	DIR_SYNC_BURST_a (s:= add_A)	DSa	Wait for the first directed reservation.
	5	record	RF	directed_start:= time at beginning of slot containing DIR_SYNC_BURST_a (s:= add_A)	DSa	Define a reference time to measure relative times from during the test.
	6	await		time= directed_start + 625 × 60/M1		
	7	send	RF	DIR_REQ_a (or:= 0; dt:= 15; nr:= 4; do:= 500; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_A) in slot beginning at time = directed_start + 625 × 60/M1	Da	Send a directed burst from ground station G, with do pointing to a slot reserved by the previous directed request, and with dt = 15 so as to cancel the directed request reservations after this superframe.
	8	rep M1		n:= 1		Verify over 1 superframe.
	9	 verify verify	 RF RF	IF n = {500, 1625, 2750, 3875} THEN DIR_SYNC_BURST_a (po= -128; pt= 0; s= add_A) in slot beginning at time = directed_start + (625 + n) × 60/M1 ELSE no transmission in slot beginning at time = directed_start + (625 + n) × 60/M1	DSa	Verify that in the reserved slots the station under test transmits bursts with po = -128 and pt = 0 thus cancelling the initial directed reservations in each of the streams. Verify transmissions are made only in the reserved slots.
	10	endrep		n:= n + 1		
postamble	11	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Autotune_CancelAbsent					
Purpose:	**To demonstrate that a station receiving a directed request addressed to another station will take no action upon receipt of a directed cancellation from the directing station alone.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 1 125; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_D)	Da	Send a directed request reservation from a simulated ground station G, requesting a simulated station D to transmit at a rate of 4 bursts per superframe for 5 superframes in the directed slots, starting in the slot do slots after the first slot of the received burst.
	5	record	RF	directed_time:= time at beginning of slot containing directed request reservation		Define a reference time to measure relative times from during the test.
	6	await		time = directed_time + 625 x 60/M1		
	7	send	RF	DIR_REQ_a (or:= 0; dt:= 15; nr:= 4; do:= 500; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_D) in slot beginning at time = directed_time + 625 x 60/M1	Da	Send a directed request reservation from station G, addressed to station D, with do pointing to a slot reserved by the previous directed request, and with dt = 15 so as to cause station D to cancel the reserved streams after this superframe.
	8	macro		M_RAND_ACC_SU (sf:= 6)		Queue random access transmissions over 6 superframes.
	9	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	
	10	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s = add_A)	Ra	Define a reference time to measure relative times from during the test.
	11	repx		n:= 1		
	12	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = start_time + n x 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots up to the reserved slot.
	13	until		time = directed_time + 1 124 x 60/M1 in previous step; n:= n + 1		End loop before first directed reservation.
	14	rep 5 x M1		n:= 0		Verify over the 5 superframes containing the directed reservations.
	15	 verify verify	 RF RF	**IF** n = {0, 1 125, 2 250, 3 375, 4 500, 5 625, 6 750, 7 875, 9 000, 10 125, 11 250, 12 375, 13 500, 14 625, 15 750, 16 875, 18 000, 19 125, 20 250, 21 375} **THEN** no transmission present in slot beginning at time = directed_time + (1 125 + n) x 60/M1 **ELSE** RAND_ACC_DATA_a (s = add_A) in slot beginning at time = directed_time + (1 125 + n) x 60/M1)	Ra	Verify that no transmissions are made by the station under test in slots originally reserved by the directed request. The reserved slots are given by do + k x (M1/nr) + j x M1 after the first slot of the received burst for j = 0 to dt and k = 0 to nr - 1. Verify that random access transmissions are made by the station under test in all slots except the reserved slots.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	16	endrep		n:= n + 1		
postamble	17	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Autotune_Override**					
Purpose:	**To demonstrate that a station receiving a directed request with the override flag set will cancel previously placed directed request reservations made by the same station.**					
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.
test body	4	send	RF	DIR_REQ_a (or:= 0; dt:= 4; nr:= 4; do:= 1125; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_D)	Da	Send a directed request reservation from a simulated ground station G, requesting a simulated station D to transmit at a rate of 4 bursts per superframe for 5 superframes in the directed slots, starting in the slot do slots after the first slot of the received burst.
	5	record	RF	directed_time:= time at beginning of slot containing directed request reservation		Define a reference time to measure relative times from during the test.
	6	await		time = directed_time + 625 × 60/M1		
	7	send	RF	DIR_REQ_a (or:= 1; dt:= 9; nr:= 3; do:= 1500; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_D) in slot beginning at time = directed_time + 625 × 60/M1	Da	Send a directed request reservation from ground station G to station D with or = 1, causing all previously placed directed request reservations to be cancelled and replaced with new reservations.
	8	macro		M_RAND_ACC_SU (sf:= 11)		Queue random access transmissions over 11 superframes.
	9	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	
	10	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s= add_A)	Ra	Define a reference time to measure relative times from during the test.
	11	repx		n:= 1		
	12	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = start_time + n × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots up to the reserved slot.
	13	until		time = directed_time + 2124 × 60/M1 in previous step; n:= n + 1		End loop before first directed reservation.
	14	rep 10×M1		n:= 0		Verify over the 10 superframes containing the directed reservations.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	15	 verify verify	 RF RF	IF n = {0, 1500, 3000, 4500, 6000, 7500, 9000, 10500, 12000, 13500, 15000, 16500, 18000, 19500, 21000, 22500, 24000, 25500, 27000, 28500, 30000, 31500, 33000, 34500, 36000, 37500, 39000, 40500, 42000, 43500} THEN no transmission present in slot beginning at time = directed_time + (2125 + n) × 60/M1 ELSE RAND_ACC_DATA_a (s= add_A) in slot beginning at time = directed_time + (2125 + n) × 60/M1	Ra	Verify that no transmissions are made by the station under test in slots reserved by the latter directed request but that transmissions are made in slots reserved by the former directed request. The reserved slots are given by do + k × (M1/nr) + j × M1 after the first slot of the received burst for j = 0 to dt and k = 0 to nr - 1. Verify that random access transmissions are made by the station under test in all slots except the reserved slots.
	16	endrep		n:= n + 1		
postamble	17	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Autotune_Invalid_A					
Purpose:	**To demonstrate that an invalid directed rate autotune reservation results in no action.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	rep 7		n:= 1		
	3	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst to be transmitted on GSC1.
	4	record	RF (GSC1)	IF n = 1 or 7 THEN sync_time(n):= time at beginning of slot containing SYNC_BURST_c	Sc	
	5	endrep		n:= n + 1		
	6	record		period1a:= (sync_time(7) - sync_time(1)) / 6		Record the average report period.
	7	rep 7		n:= 1		
	8	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst to be transmitted on GSC2.
	9	record	RF (GSC2)	IF n = 1 or 7 THEN sync_time(n):= time at beginning of slot containing SYNC_BURST_c	Sc	
	10	endrep		n:= n + 1		
	11	record		period2a:= (sync_time(7) - sync_time(1)) / 6		Record the average report period on GSC2.
	12	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 14; nr:= 4; do:= 0; lg:= 0; f:= frequency of GSC2; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst from a simulated ground station G, requesting the station under test to autonomously transmit at a rate of 4 bursts per superframe for 14 superframes.
	13	rep 7		n:= 1		
	14	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst to be transmitted on GSC1.
	15	record	RF (GSC1)	IF n = 1 or 7 THEN sync_time(n):= time at beginning of slot containing SYNC_BURST_c	Sc	
	16	endrep		n:= n + 1		
	17	record		period1b:= (sync_time(7) - sync_time(1)) / 6		Record the average report period.
	18	rep 7		n:= 1		
	19	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst to be transmitted on GSC2.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	20	record	RF (GSC2)	IF n = 1 or 7 THEN sync_time(n):= time at beginning of slot containing SYNC_BURST_c	Sc	
	21	endrep		n:= n + 1		
	22	record		period2b:= (sync_time(7) - sync_time(1)) / 6		Record the average report period on GSC2.
	23	verify		period1a = period1b AND period2a = period2b		Verify that the frequencies on GSC1 and GSC2 remain the same following the autotune command.
postamble	24					Bring test equipment into idle state.
Comments:						

Test Case Name:	**Autotune_Invalid_B**					
Purpose:	**To demonstrate that an invalid frequency field in an autotune reservation results in no reservation.**					
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.
test body	4	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 0; nr:= 4; do:= 10; lg:= 0; f:= 000 hex; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst from a simulated ground station G, requesting the station under test to transmit at a rate of 4 bursts per superframe for 1 superframe.
	5	record	RF	dir_time:= time at beginning of slot containing DIR_REQ_a	Da	
	6	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	7	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	8	rep M1		p:= 0		
	9	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = dir_time + p × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots following the autotune command.
	10	endrep		p:= p + 1		
	11	send	RF (GSC1)	DIR_REQ_a (or:= 0; dt:= 0; nr:= 4; do:= 10; lg:= 0; f:= 500 hex; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G (address indicating source is a ground station); d:= add_A)	Da	Send a directed burst from a simulated ground station G, requesting the station under test to transmit at a rate of 4 bursts per superframe for 1 superframe.
	12	record	RF	dir_time:= time at beginning of slot containing DIR_REQ_a	Da	
	13	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	14	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	15	rep M1		p:= 0		
	16	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = dir_time + p × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots following the autotune command.
	17	endrep		p:= p + 1		
postamble	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	19	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
Comments:						

Test Case Name:	Autotune_DirectedCessation					
Purpose:	**To demonstrate that on cessation of an autotune command directing slots to use, normal default behaviour on the GSCs is established.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	send	RF	DIR_REQ_a (or:= 0; dt:= 0; nr:= 60; do:= 10; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_D)	Da	Send a directed slot reservation from a simulated ground station G, requesting a simulated station D to transmit at a rate of 60 bursts per superframe for 1 superframe in the directed slots, starting in the slot do slots after the first slot of the received burst.
	3	record	RF	dir_time:= time at beginning of slot containing DIR_REQ_a (s= add_G)	Da	Define a reference time for the directed slot burst.
	4	await		time:= dir_time + 10 × 60/M1		
	5	rep 60		n:= 1		
	6	verify		SYNC_BURST_c (s= add_A) in slot beginning at time:= dir_time + (10 + n × 75) × 60/M1	Sc	
	7	endrep		n:= n + 1		
	8	await	RF	SYNC_BURST_c (s= add_A)	Sc	Await next autonomous sync burst following expiry of the autotune command.
	9	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_c (s= add_A)	Sc	Define a reference time to measure relative times from during the test.
	10	rep 10		n:= 1		Repeat test on GSC1 10 times.
	11	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	
	12	record	RF	IF n = 10 THEN time(n) = time at the beginning of slot containing SYNC_BURST_c (s= add_A)	Sc	
	13	endrep		n:= n + 1		
	14	verify		time(10) - sync_time ≤ 10 × M1		Verify that on GSC1 the station emits at least 1 sync burst per M1 slots.
	15	rep 10		n:= 1		Repeat test on GSC2 10 times.
	16	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	
	17	record	RF	IF n = 10 THEN time(n) = time at the beginning of slot containing SYNC_BURST_c (s= add_A)	Sc	
	18	endrep		n:= n + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	19	verify		time(10) - sync_time ≤ 10 × M1		Verify that on GSC1 the station emits at least 1 sync burst per M1 slots.
postamble	20					Bring test equipment into idle state.
Comments:						

Test Case Name:	**Autotune_DirectedCancelAutonomous**					
Purpose:	**To demonstrate that receipt of a directed slot autotune causes autonomous reporting of the same data to be cancelled.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	rep 6		n:= 1		
	3	await		SYNC_BURST_c (s= add_A)	Sc	
	4	record	RF	auto_time(n):= time at beginning of slot containing SYNC_BURST_c (s= add_A)	Sc	
	5	endrep		n:= n + 1		
	6	send	RF	DIR_REQ_a (or:= 0; dt:= 1; nr:= 3; do:= 2; lg:= 0; f:= 0; rcvr:= 0; trmt:= 0; r-mi:= 0; s:= add_G; d:= add_D)	Da	Send a directed slot reservation from a simulated ground station G, requesting a simulated station D to transmit at a rate of 3 bursts per superframe for 2 superframes in the directed slots, starting in the slot do slots after the first slot of the received burst.
	7	rep 6		n:= 1		
	8	await		time:= auto_time(n) + 60		
	9	verify	RF	po= 0 and pt= 0 in SYNC_BURST_c (s= add_A) in slot beginning at time:= auto_time(n) + 60	Sc	Verify that the existing autonomous sync burst streams are cancelled.
	10	endrep		n:= n + 1		
postamble	11					Bring test equipment into idle state.
Comments:						

Test Case Name:	Autotune_DirectedQuarantine					
Purpose:	**To demonstrate that a response to a directed request issued by a ground station will cause the following slots to be regarded as quarantined by a station receiving the response.**					
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; VS1:= 2)		Ensure 100 % chance of transmission on access VS1 slots reserved as quarantined slots after the ground station periodic reservation.
test body	4	send	RF	DIR_SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B)	DSa	Send a directed sync burst (burst length 1, a/d flag set to 1) from a simulated station B, reserving the same transmission slot, and the following VS1 ground quarantined slots, in the next 4 superframes.
	5	record	RF	periodic_start:= time at beginning of slot containing the sync burst		Provides a reference time for the directed burst.
	6	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 5 superframes.
	7	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	8	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s= add_A)	Ra	
	9	repx		n:= 1		
	10	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time =start_time + n × 60/M1	Ra	Verify that random access transmissions are made by the station under test in slots preceding the reserved slot.
	11	until		time = periodic_start + (M1 - 1) × 60/M1 in previous step; n:= n + 1		End loop in slot immediately preceding reserved slot.
	12	rep 4 × M1		n:= 0		Verify over 4 superframes.
	13	 verify verify	 RF RF	IF n = {0, 1, 2, M1, M1 + 1, M1 + 2, (2 × M1), (2 × M1) + 1, (2 × M1) + 2, (3 × M1), (3 × M1) + 1, (3 × M1) + 2} THEN no transmission present in slot beginning at time = periodic_start + (n + M1) × 60/M1 ELSE RAND_ACC_DATA_a (s= add_A) in slot beginning at time = periodic_start + (n + M1) × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the reserved slots and the following ground quarantined slots.
	14	endrep		n:= n + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
postamble	15	send	VSS	SET PARAMETERS (p:= 64/256; VS1:= 4)		Reset to default values.
	16	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	PleaResponse_Reservation_A					
Purpose:	**To demonstrate that receipt of a plea response with a standard nominal rate causes the appropriate slots to be reserved.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	PLEA_RESP_a (a_1:= 1; a_2:= 2; a_3:= 3; a_4 to a_{11}:= 0; nr:= 2; off:= 10; s = add_B; d = add_C)	PRa	Send a plea response from a simulated station B to a simulated station C with nr ≠ special. The burst reserves an initial slot 10 slots after the transmission slot followed by three additional slots.
	5	record	RF	plea_time:= time at beginning of slot containing PLEA_RESP_a	PRa	
	6	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	8	rep M1		p:= 0		
	9	verify	RF	**IF** p = {10, 2 261, 4 512, 6 763} **THEN** No RAND_ACC_DATA_a (s = add_A) in slot beginning at time = plea_time + p x 60/M1	Ra	Verify that no random access transmissions are made by the station under test in slots reserved by the plea response.
	10	endrep		p:= p + 1		
	11	send	RF	PLEA_RESP_a (a_1:= 15; a_2:= 30; a_3 to a_{11}:= 0; nr:= 3; off:= 100; s = add_B; d = add_C)	PRa	Send a plea response from a simulated station B to a simulated station C with nr ≠ special. The burst reserves an initial slot 100 slots after the transmission slot followed by two additional slots.
	12	record	RF	plea_time:= time at beginning of slot containing PLEA_RESP_a	PRa	
	13	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	14	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	15	rep M1		p:= 0		
	16	verify	RF	**IF** p = {100, 1 615, 3 130} **THEN** No RAND_ACC_DATA_a (s = add_A) in slot beginning at time = plea_time + p x 60/M1	Ra	Verify that no random access transmissions are made by the station under test in slots reserved by the plea response.
	17	endrep		p:= p + 1		
postamble	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	19	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
Comments:						

Test Case Name:	PleaResponse_Reservation_B					
Purpose:	**To demonstrate that receipt of a plea response with a special nominal rate causes the appropriate slots to be reserved.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	PLEA_RESP_b (a_1:= 100; a_2:= 200; a_3:= 300; a_4 to a_5:= 0; nr:=1111binary; off:= 500; s = add_B; d = add_C)	PRb	Send a plea response from a simulated station B to a simulated station C with nr = special. The burst reserves an initial slot 500 slots after the transmission slot followed by three slots.
	5	record	RF	plea_time:= time at beginning of slot containing PLEA_RESP_b	PRb	
	6	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	7	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	8	rep M1		p:= 0		
	9	verify	RF	**IF** p = {500, 600, 800, 1100} **THEN** No RAND_ACC_DATA_a (s = add_A) in slot beginning at time = plea_time + p x 60/M1	Ra	Verify that no random access transmissions are made by the station under test in slots reserved by the plea response.
	10	endrep		p:= p + 1		
	11	send	RF	PLEA_RESP_b (a_1:= 250; a_2:= 750; a_3 to a_5:= 0; nr:=1111binary; off:= 150; s = add_B; d = add_C)	PRb	Send a plea response from a simulated station B to a simulated station C with nr = special. The burst reserves an initial slot 150 slots after the transmission slot followed by two slots.
	12	record	RF	plea_time:= time at beginning of slot containing PLEA_RESP_b	PRb	
	13	macro		M_RAND_ACC_SU (sf:= 1)		Queue random access transmissions over 1 superframe.
	14	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	15	rep M1		p:= 0		
	16	verify	RF	**IF** p = {150, 400, 1150} **THEN** No RAND_ACC_DATA_a (s = add_A) in slot beginning at time = plea_time + p x 60/M1	Ra	Verify that no random access transmissions are made by the station under test in slots reserved by the plea response.
	17	endrep		p:= p + 1		
postamble	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
	19	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
Comments:						

Test Case Name:	**PleaResponse_Transmission_A**					
Purpose:	**To demonstrate that receipt of a plea addressed to a station results in transmission of a plea response of the appropriate format.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	RF	PLEA_a (s = add_B; d = add_A)	Pa	Send a plea transmission from a simulated station B to the station under test.
	4	record	RF	plea_time:= time at beginning of slot containing PLEA_a	Pa	
	5	await		time = plea_time + 2		Wait for TL5 s.
	6	verify	RF	PLEA_RESP_a (s = add_A; d = add_B) with $a_1 \neq 0$ **OR** PLEA_RESP_b (s = add_A; d = add_B) with $a_1 \neq 0$ transmitted before time:= plea_time + 2	PRa, PRb	Verify that a plea response is issued by the station under test addressed to station B within TL5 s.
postamble	7	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**PleaResponse_Transmission_B**					
Purpose:	**To demonstrate that a second plea addressed to a station results in transmission of a plea response containing the remaining future slots from the previous plea response.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	RF	PLEA_a (s = add_B; d = add_A)	Pa	Send a plea transmission from a simulated station B to the station under test.
	4	await	RF	PLEA_RESP_a (s = add_A; d = add_B) with $a_1 \neq 0$ **OR** PLEA_RESP_b (s = add_A; d = add_B) with $a_1 \neq 0$	PRa, PRb	Wait for the plea response issued by the station under test addressed to station B.
	5	record	RF	pr_time:= time at beginning of slot containing PLEA_RESP_a **OR** PLEA_RESP_b	PRa, PRb	
	6	record	RF	LIST1:= list of slot reservations provided in PLEA_RESP_a **OR** PLEA_RESP_b	PRa, PRb	
	7	await		time = pr_time + 30		Wait for half a superframe.
	8	send	RF	PLEA_a (s = add_B; d = add_A)	Pa	Send a second plea transmission from a simulated station B to the station under test.
	9	await	RF	PLEA_RESP_a (s = add_A; d = add_B) with $a_1 \neq 0$ **OR** PLEA_RESP_b (s = add_A; d = add_B) with $a_1 \neq 0$	PRa, PRb	Wait for the second plea response issued by the station under test addressed to station B.
	10	record	RF	LIST2:= list of slot reservations provided in PLEA_RESP_a **OR** PLEA_RESP_b	PRa, PRb	

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	11	verify	RF	Remaining reservations in LIST1 are included in LIST2		Verify that all remaining reservations provided in the first plea response are included in the second plea response.
postamble	12	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**PleaResponse_Retransmission**					
Purpose:	**To demonstrate that a plea response is not re-transmitted.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	RF	PLEA_a (s = add_B; d = add_A)	Pa	Send a plea transmission from a simulated station B to the station under test.
	4	await	RF	PLEA_RESP_a (s = add_A; d = add_B) with $a_1 \neq 0$ **OR** PLEA_RESP_b (s = add_A; d = add_B) with $a_1 \neq 0$	PRa, PRb	Wait for the plea response is issued by the station under test addressed to station B.
	5	record	RF	pr_time:= time at beginning of slot containing PLEA_RESP_a **OR** PLEA_RESP_b	PRa, PRb	
	6	await		time = pr_time + 60		Wait for one superframe.
	7	verify	RF	No re-transmission of PLEA_RESP_a **OR** PLEA_RESP_b by station under test	PRa, PRb	Verify that no re-transmission of the plea response occurs.
postamble	8	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Block_Superframe_Reserve_A					
Purpose:	**To demonstrate that the station will not transmit in blocks of slots reserved by a ground station by means of a superframe block message.**					
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.
test body	4	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; s:= add_G; lat:= CPR_LAT(0); lon:= CPR_LON(E 150 NM)) (position of station G is < VS4 away from station under test)	Sa	Send a sync burst from a simulated ground station G < VS4 away from the station under test, reporting G's position.
	5	send	RF	SUPER_BLOCK_a (bt:= 2; br:= 2; roff:= 17; bs:= 20; bo:= 0; blg:= 2; s:= add_G; d:= add_B)	SUa	Send a burst containing a superframe block reservation from ground station G to a simulated mobile B. The first reserved block starts bs = 20 slots after the transmission slot. A second reserved block is set to start half a superframe later due to the block repeat rate (br) of 2 per superframe. Block length equals 3. The re-broadcast offset roff = 17 reserves a slot for re-broadcast of the block reservation by mobile B. The transmission slot, re-broadcast slot, and the reserved blocks are also reserved for a further two superframes as set by the block timeout bt = 2.
	6	record	RF	block_start:= time at beginning of slot containing SUPER_BLOCK_a reservation	SUa	Provides a reference time for the blocks reserved by the ground station G.
	7	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 5 superframes.
	8	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	9	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s= add_A)	Ra	
	10	repx		n:= 1		
	11	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = start_time + n × 60/M1	Ra	Verify that random access transmissions are made by the station under test in slots preceding the reserved blocks.
	12	until		time = block_start + 16 × 60/M1 in previous step; n:= n + 1		End loop in slot immediately preceding re-broadcast slot.
	13	rep 4×M1		n:= 0		Verify over 4 superframes.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14			IF n = {17, 20, 21, 22, M1/2 + 20, M1/2 + 21, M1/2 + 22, M1, M1 + 17, M1 + 20, M1 + 21, M1 + 22, M1 + M1/2 + 20, M1 + M1/2 + 21, M1 + M1/2 + 22, 2 × M1, 2 × M1 + 17, 2 × M1 + 20, 2 × M1 + 21, 2 × M1 + 22, 2 × M1 + M1/2 + 20, 2 × M1 + M1/2 + 21, 2 × M1 + M1/2 + 22} THEN	Ra	Verify that random access transmissions are made by the station under test in all slots except for the reserved blocks, repeat block reservation slots, and repeat re-broadcast slots.
		verify	RF	no transmission present in slot beginning at time = block_start + n × 60/M1 ELSE		
		verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = block_start + n × 60/M1		
	15	endrep		n:= n + 1		
postamble	16	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	17	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Block_Superframe_Reserve_B					
Purpose:	**To demonstrate that the station will not transmit in blocks of slots reserved by a ground station by means of a superframe block message with block offset (bo) non-zero but block timeout (bt) set to 3.**					
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.
test body	4	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; s:= add_G; lat:= CPR_LAT(0); lon:= CPR_LON(E 150 NM)) (position of station G is < VS4 away from station under test)	Sa	Send a sync burst from a simulated ground station G < VS4 away from the station under test, reporting G's position.
	5	send	RF	SUPER_BLOCK_a (bt:= 3; br:= 2; roff:= 17; bs:= 20; bo:= 5; blg:= 1; s:= add_G; d:= add_B)	SUa	Send a burst containing a superframe block reservation from ground station G to a simulated mobile B. The first reserved block starts bs = 20 slots after the transmission slot. A second reserved block is set to start half a superframe later due to the block repeat rate (br) of 2 per superframe. Block length equals 2. The re-broadcast offset roff = 17 reserves a slot for re-broadcast of the block reservation by mobile B. The transmission slot, re-broadcast slot, and the reserved blocks are also reserved for a further three superframes as set by the block timeout bt = 3. The non-zero block offset value should not affect the future reservations as bt = 3.
	6	record	RF	block_start:= time at beginning of slot containing SUPER_BLOCK_a reservation	SUa	Provides a reference time for the blocks reserved by the ground station G.
	7	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 5 superframes.
	8	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	9	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s= add_A)	Ra	
	10	repx		n:= 1		
	11	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = start_time + n × 60/M1	Ra	Verify that random access transmissions are made by the station under test in slots preceding the reserved blocks.
	12	until		time = block_start + 16 × 60/M1 in previous step; n:= n + 1		End loop in slot immediately preceding re-broadcast slot.
	13	rep 5×M1		n:= 0		Verify over 5 superframes.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14			IF n = {17, 20, 21, M1/2 + 20, M1/2 + 21, M1, M1 + 17, M1 + 20, M1 + 21, M1 + M1/2 + 20, M1 + M1/2 + 21, 2 × M1, 2 × M1 + 17, 2 × M1 + 20, 2 × M1 + 21, 2 × M1 + M1/2 + 20, 2 × M1 + M1/2 + 21, 3 × M1, 3 × M1 + 17, 3 × M1 + 20, 2 × M1 + 21, 3 × M1 + M1/2 + 20, 3 × M1 + M1/2 + 21} THEN	Ra	Verify that random access transmissions are made by the station under test in all slots except for the reserved blocks, repeat block reservation slots, and repeat re-broadcast slots.
		verify	RF	no transmission present in slot beginning at time = block_start + n × 60/M1 ELSE		
		verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = block_start + n × 60/M1		
	15	endrep		n:= n + 1		
postamble	16	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	17	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Block_Superframe_Reserve_C**					
Purpose:	**To demonstrate that the station will not transmit in blocks of slots reserved by a ground station by means of a superframe block message with block offset (bo) non-zero.**					
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.
test body	4	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; s:= add_G; lat:= CPR_LAT(0); lon:= CPR_LON(E 150 NM)) (position of station G is < VS4 away from station under test)	Sa	Send a sync burst from a simulated ground station G < VS4 away from the station under test, reporting G's position.
	5	send	RF	SUPER_BLOCK_a (bt:= 2; br:= 2; roff:= 17; bs:= 20; bo:= 5; blg:= 1; s:= add_G; d:= add_B)	SUa	Send a burst containing a superframe block reservation from ground station G to a simulated mobile B. The first reserved block starts bs = 20 slots after the transmission slot. A second reserved block is set to start half a superframe later due to the block repeat rate (br) of 2 per superframe. Block length equals 2. The re-broadcast offset roff = 17 reserves a slot for re-broadcast of the block reservation by mobile B. The transmission slot, re-broadcast slot, and the reserved blocks are also reserved for a further two superframes as set by the block timeout bt = 2. The block offset value bo = 5, indicating that reservations for the blocks, the reservation slot, and the re-broadcast slot, are offset by 5 slots after the third superframe.
	6	record	RF	block_start:= time at beginning of slot containing the block reservation		Provides a reference time for the blocks reserved by the ground station G.
	7	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 5 superframes.
	8	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	9	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s= add_A)	Ra	
	10	repx		n:= 1		
	11	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = start_time + n × 60/M1	Ra	Verify random access transmissions are made by the station under test in slots preceding the reserved blocks.
	12	until		time = block_start + 16 × 60/M1 in previous step; n:= n + 1		End loop in slot immediately preceding re-broadcast slot.
	13	rep 5×M1		n:= 0		Verify over 5 superframes.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	 verify verify	 RF RF	IF n = {17, 20, 21, M1/2 + 20, M1/2 + 21, M1, M1 + 17, M1 + 20, M1 + 21, M1 + M1/2 + 20, M1 + M1/2 + 21, 2 × M1, 2 × M1 + 17 2 × M1 + 20, 2 × M1 + 21, 2 × M1 + M1/2 + 20, 2 × M1 + M1/2 + 21, 3 × M1 + 5, 3 × M1 + 22, 3 × M1 + 25, 2 × M1 + 26, 3 × M1 + M1/2 + 25, 3 × M1 + M1/2 + 26} THEN no transmission present in slot beginning at time = block_start + n × 60/M1 ELSE RAND_ACC_DATA_a (s= add_A) in slot beginning at time = block_start + n × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except for the reserved blocks, repeat block reservation slots, repeat re-broadcast slots, and the offset positions of these in the fourth superframe.
	15	endrep		n:= n + 1		
postamble	16	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	17	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Block_Superframe_Invalid					
Purpose:	**To demonstrate that the station will ignore a superframe block reservation with re-broadcast offset < 2.**					
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.
test body	4	send	RF	SYNC_BURST_a (pt:= 1; po:= 0; s:= add_G; lat:= CPR_LAT(0); lon:= CPR_LON(E 150 NM)) (position of station G is < VS4 away from station under test)	Sa	Send a sync burst from a simulated ground station G < VS4 away from the station under test, reserving the same transmission slot in the next superframe.
	5	record	RF	sync_start:= time at beginning of slot containing the SYNC_BURST_a reservation	Sa	Provides a reference time for the reservation by the ground station G.
	6	macro		M_RAND_ACC_SU (sf:= 5)		Queue random access transmissions over 5 superframes.
	7	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	8	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s= add_A)	Ra	
	9	repx		n:= 1		
	10	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = start_time + n × 60/M1	Ra	Verify that random access transmissions are made by the station under test in slots preceding the reserved slot.
	11	until		time = sync_start + (M1 - 1) × 60/M1 in previous step; n:= n + 1		End loop in slot immediately preceding reserved slot.
	12	send	RF	SUPER_BLOCK_a (bt:= 2; br:= 2; roff:= 1; bs:= 2; bo:= 0; blg:= 1; s:= add_G; d:= add_B) in slot beginning at time = sync_start + 60	SUa	Send a burst containing a superframe block reservation from a simulated ground station G to a simulated mobile B with an invalid re-broadcast offset value, roff = 1.
	13	rep 4×M1		n:= 1		Verify over 4 superframes.
	14	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = sync_start + (M1 + n) × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots after the block reservation over 4 superframes.
	15	endrep		n:= n + 1		
postamble	16	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	17	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Block_Superframe_Rebroadcast**					
Purpose:	**To demonstrate that a station will re-broadcast a superframe blocking message in accordance with a ground station request.**					
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.
test body	4	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_G; lat:= CPR_LAT(0); lon:= CPR_LON(E 150 NM)) (position of station G is < VS4 away from station under test)	Sa	Send a sync burst from a simulated ground station G < VS4 away from the station under test, reporting G's position.
	5	send	RF	SUPER_BLOCK_a (bt:= 2; br:= 3; roff:= 17; bs:= 20; bo:= 5; blg:= 2; s:= add_G; d:= add_A)	SUa	Send a burst containing a superframe block reservation from ground station G addressed to the station under test. The first reserved block starts bs = 20 slots after the transmission slot. Block length equals 3. The re-broadcast offset roff = 17 reserves a slot for re-broadcast of the block reservation by mobile B.
	6	record	RF	block_start:= time at beginning of slot containing the block reservation		Provides a reference time for the blocks reserved by the ground station G.
	7	await		time = block_start + 17 × 60/M1		End loop in slot immediately preceding re-broadcast slot.
	8	verify	RF	RE_BROADCAST_a (bt:= 2; br:= 3; roff:= 3; bs:= 3; bo:= 5; blg:= 2; s:= add_A) in slot beginning at time = block_start + 17 × 60/M1	RBa	Verify that the message is re-broadcast in accordance with ICAO VDL SARPs [1] requirements. These are: br, bo, bt set to the same as the received burst; bs to indicate the same first slot of the first block; roff set to equal bs; no destination subfield.
postamble	9	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Block_Superframe_Quarantine**					
Purpose:	**To demonstrate that a block reservation message issued by a station greater than distance VS4 away will not cause slots to be reserved.**					
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.
test body	4	rep 4		n:= {30, 20, 0, 100}		Repeat for different values of the VS4 parameter.
	5	send	VSS	SET PARAMETERS (VS4:= n)		
	6	send	RF	SYNC_BURST_a (pt:= 0; po:= 0; a/d:= 1; s:= add_G, address indicating source is a ground station; lat:= CPR_LAT(0); lon:= CPR_LON(E (n + 10) NM)) (position of ground station G is > VS4 away from station under test)	Sa	Send a sync burst from a simulated ground station G > VS4 away from the station under test, reporting the ground station's position.
	7	send	RF	SUPER_BLOCK_a (bt:= 2; br:= 2; roff:= 17; bs:= 20; bo:= 0; blg:= 2; s:= add_G; d:= add_B)	SUa	Send a burst containing a superframe block reservation from ground station G to a simulated mobile B. The first reserved block starts bs = 20 slots after the transmission slot. A second reserved block is set to start half a superframe later due to the block repeat rate (br) of 2 per superframe. Block length equals 3. The re-broadcast offset roff = 17 reserves a slot for re-broadcast of the block reservation by mobile B. The transmission slot, re-broadcast slot, and the reserved blocks are also reserved for a further two superframes as set by the block timeout bt = 2.
	8	record	RF	block_start:= time at beginning of slot containing the block reservation		Provides a reference time for the blocks reserved by the ground station G.
	9	macro		M_RAND_ACC_SU (sf:= 4)		Queue random access transmissions over 4 superframes.
	10	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	11	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s= add_A)	Ra	
	12	rep 4×M1		p:= 0		
	13	verify	RF	RAND_ACC_DATA_a (s= add_A) in slot beginning at time = start_time + p × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots, including those reserved by the block reservation, over 4 superframes.
	14	endrep		p:= p + 1		
	15	endrep		next n		
postamble	16	send	VSS	SET PARAMETERS (p:= 64/256; VS4:= 300 NM)		Reset to default values.
	17	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Block_Secondframe_Reserve					
Purpose:	**To demonstrate that the station will not transmit in blocks of slots reserved by a ground station by means of a second frame block message.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	verify	RF	Transceiver is tuned either to GSC1 or to GSC2.		Ensure that the transceiver is operating on either one of the GSCs.
	3	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
	4	send	VSS	SET PARAMETERS (p:= 1, V66:= 8)		Ensure 100 % chance of transmission on access. Set V66 to its default value.
test body	5	rep 4		n:= {8, 4, 0, 31}		Repeat for different values of the V66 parameter.
	6	send	RF	SECOND_BLOCK_a (vt:= 1; sz:= n; s:= add_G)	SCa	Send a general request burst containing a second frame block reservation from ground station G. The burst reserves the first sz = n slots of every UTC second for ground station use for a period of vt = 1 superframe.
	7	rep 100		k:= 1		Repeat over 100 s.
	8	macro		M_RAND_ACC_SL (slots:= 75)		Queue random access transmissions over 75 slots (1 s).
	9	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	10	await		Start of next UTC second		Wait for the start of a UTC second.
	11	record		utc_start:= time at start of UTC second		
	12	rep 75		p:= 0		
	13	 verify verify	 RF RF	IF p < n THEN no transmission present in slot beginning at time = utc_start + p × 60/M1 ELSE RAND_ACC_DATA_a (s= add_A) in slot beginning at time = utc_start + p × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the n blocked slots at the start of each UTC second.
	14	endrep		p:= p + 1		
	15	endrep		k:= k + 1		
	16	endrep		next n		
postamble	17	send	VSS	SET PARAMETERS (p:= 64/256; V66:= 8; TV61:= 4)		Reset to default values.
	18	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Response_Reservation**					
Purpose:	**To demonstrate that a response reservation field is recognized and causes no reservation to be made.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station 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.
test body	4	send	RF	SYNC_BURST_m (s:= add_B; d:= 7)	Sm	Send a sync burst with a response reservation from a simulated station B to the station under test, with the destination address equal to 7, indicating a broadcast burst with an equivalent to a null reservation (see note).
	5	macro		M_RAND_ACC_SU (sf:= 4)		Queue random access transmissions over 4 superframes.
	6	await	RF	RAND_ACC_DATA_a (s = add_A)	Ra	Wait for the start of the random access transmissions.
	7	record	RF	start_time:= time at beginning of slot containing RAND_ACC_DATA_a (s = add_A)	Ra	
	8	rep 4xM1		n:= 0		
	9	verify	RF	RAND_ACC_DATA_a (s = add_A) in slot beginning at time = start_time + n x 60/M1	Ra	Verify that random access transmissions are made by the station under test in consecutive slots for 4 superframes, and therefore that no reservation was made by the response reservation.
	10	endrep		n:= n + 1		
postamble	11	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default values.
	12	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						
NOTE: The destination address d set to 7 implies that bits 1 through 24 of the destination subfield d are absent, and that bits 25 to 27 are set to 111 binary.						

Test Case Name:	**Request_Unsupported**					
Purpose:	**To demonstrate that a station will respond to a general request burst that cannot be supported with a general failure burst.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the ground station for testing.
	2	send	VSS	SUPPRESS AUTONOMOUS SYNC BURSTS		Suppress the autonomous sync bursts to avoid possible confliction.
test body	3	send	RF	UNI_BURST_d (ro:= 100; lg:= 0; sdf:= 0; pr:= 1; r-mi:= 1010101; s:= add_G; d:= add_A)	Ud	Send a general request burst from a simulated ground station G, addressed to the station under test, with the requested message ID set to 1010101 binary which is reserved for future use and therefore not supported.
	4	verify	RF	GEN_RESP_a (s:= add_A; d:= add_G; ok = 0; r-mi = 1010101; bd = FF hex; err = 00 hex)	GRa	Verify that the station under test responds with a general response, with ok = 0 indicating a general failure, to a general request that cannot be supported.
postamble	5	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments: Certain stations, such as those with low power (Type B) transmitters, will not support a general request for a sync burst using a unicast reservation. For such stations, this test is inapplicable.						

Test Case Name:	**Sync_Format**					
Purpose:	**To demonstrate that an autonomous sync burst is emitted 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}		Start loop. Identify a set of parameters to use.
	3	send	AppIn	REQUEST TO TRANSMIT SYNC BURST (SYNC_BURST_APPIN_PARAMETERS(x))		Instruct the station under test to transmit a sync burst.
	4	await	RF	BURST broadcast by the station under test		Wait for the sync burst to be transmitted by the station under test.
	5	verify	RF	BURST (s = add_A) has the format of 'sYNC_BURST_l"	SI	Verify that the transmitted burst is the same as 'sYNC_BURST_l" set out in burst format clause.
	6	verify	RF	SYNC_BURST_RF_OUT_PARAMETERS(x)		Verify that the content of the burst is correct.
	7	endrep		next x		Select next set of parameters.
postamble	8					Bring test equipment into idle state.
Comments:						

Test Case Name:	**Sync_Format_Rec**					
Purpose:	**To demonstrate that a station will correctly process a received 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}		Start loop. Identify a set of parameters to use.
	3	send	RF	SYNC_BURST_l (SYNC_BURST_RF_PARAMETERS(x))	SI	Send a sync burst to the station under test from a simulated station B.
	4	await	AppOut	SYNC_BURST INFORMATION		Wait for the message information to be output by station A at the AppOut PCO.
	5	verify	AppOut	SYNC_BURST_APPOUT_PARAMETERS(x)		Verify that the content of the information output by station A at the AppOut PCO is correct.
	6	endrep		next x		Select next set of parameters.
postamble	7					Bring test equipment into idle state.
Comments:						

Test Case Name:	**Sync_Latency**					
Purpose:	**To demonstrate that the field data age of the synchronization burst is correctly set.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	send	VSS	SET PARAMETERS ($TV11_{min}$:= 8; V11:= 10) associated with sync burst generation		$TV11_{max}$ equals 8 by default. $TV11_{min}$ set to 8 to cause dither after 8 superframes. V11 set to 10 bursts within M1 slots.
	3	send	Position	Input position parameters as: lat:= 0; lon:= E 21 NM		Send (Position) initial position data.
	4	await	RF	SYNC_BURST_I (s= add_A; lat:= CPR_LAT(0); lon:= CPR_LON(E 21 NM))	SI	
	5	verify	RF	lat = CPR_LAT(0) and lon = CPR_LON(E (21) appear in SYNC_BURST_I	SI	Verify (RF) that correct position data appears in burst.
	6	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_I (s= add_A)	SI	Define a reference time to measure relative times from during the test.
	7	rep 50		n:= 1		Repeat test 50 times.
	8	await		time = sync_time + n × 6 - 0,05 - 0,1 × (n - 1)		Wait until dt before updating position data. The length of dt begins at 50 ms and is subsequently increased in 100 ms steps.
	9	send	Position	Update position parameters to: lat:= n NM; lon:= CPR_LON(E (21 + n) NM		Send (Position) revised position data.
	10	await	RF	SYNC_BURST_I (s= add_A) at time = sync_time + n × 6	SI	
	11	verify	RF	lat = CPR_LAT(n) and lon = CPR_LON(E (21 + n) appear in SYNC_BURST_I	SI	Verify (RF) that revised position data appears in burst.
	12	record	RF	DA(n):= da of SYNC_BURST_I	SI	Record data age (latency) given for data in sync burst.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	13	 verify verify verify verify verify verify verify	 RF RF RF RF RF RF RF	FOR n ≤ 10: da = n - 1 FOR n > 10: IF n = 11 or n = 12 THEN da = 10 IF n = 13 or n = 14 or n = 15 THEN da = 11 IF n = 16 to 20 THEN da = 12 IF n = 21 to 30 THEN da = 13 IF n = 31 to 40 THEN da = 14 FOR n > 40: da = 15		Verify that the values of da given in the sync burst agree with the actual values (see table 1-69 in the ICAO VDL4 Technical Manual [1]).
	14	endrep		n:=n + 1		
postamble	15	send	VSS	SET PARAMETERS ($TV11_{min}$:= 4; V11:= 1) associated with sync burst generation		Reset to default values.
Comments:						

Test Case Name:	**Sync_Rate**					
Purpose:	**To demonstrate that the station outputs autonomous sync bursts at a rate of at least once per M1 slots on each Global Signalling Channel (GSC).**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	await	RF	SYNC_BURST_c (s= add_A)	Sc	
	3	record	RF	sync_time:= time at the beginning of slot containing SYNC_BURST_c (s= add_A)	Sc	Define a reference time to measure relative times from during the test.
	4	rep 10		n:= 1		Repeat test on GSC1 10 times.
	5	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	
	6	record	RF	IF n = 10 THEN time(n) = time at the beginning of slot containing SYNC_BURST_c (s= add_A)	Sc	
	7	endrep		n:= n + 1		
	8	verify		time(10) - sync_time ≤ 10 × M1		Verify that on GSC1 the station emits at least 1 sync burst per M1 slots.
	9	rep 10		n:= 1		Repeat test on GSC2 10 times.
	10	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	
	11	record	RF	IF n = 10 THEN time(n) = time at the beginning of slot containing SYNC_BURST_c (s= add_A)	Sc	
	12	endrep		n:= n + 1		
	13	verify		time(10) - sync_time ≤ 10 × M1		Verify that on GSC1 the station emits at least 1 sync burst per M1 slots.
postamble	14					Bring test equipment into idle state.
Comments:						

Test Case Name:	**Sync_Interval**					
Purpose:	**To demonstrate that a station outputs autonomous sync bursts with a uniform interval between nominal slots on each GSC.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS ($TV11_{min}$:= 1; $TV11_{max}$:= 1; V12:= (10/M1) × V11) associated with sync burst generation		TV11 reservation hold timer set to cause dither after every superframe. V11 equals default value of 1. V12 set to give dither range of ±5.
Test body	3	rep 55		k:= 1		Repeat test 55 times to generate statistical sample.
	4	record		n:= 2k - 1		
	5	await	RF (GSC1)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst to be transmitted on GSC1.
	6	record	RF	sync_time(n):= time at beginning of slot of n^{th} SYNC_BURST_c (s= add_A) diff_time:= sync_time(n) - sync_time(1) - (n - 1) × 30 slot_diff(n):= diff_time × M1/60	Sc	Record the time of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each slot and the slot of the first burst in the sequence and transpose to a common time reference. Convert time differences to slot differences.
	7	await	RF (GSC2)	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst to be transmitted on GSC2.
	8	record	RF	sync_time(n + 1):= time at beginning of slot of n^{th} SYNC_BURST_c (s= add_A) diff_time:= sync_time(n + 1) - sync_time(1) - n × 30 slot_diff(n):= diff_time × M1/60	Sc	Record the time of the n^{th} sync burst. sync_time(1) defines a reference time to measure relative times from during the test. Calculate the relative time differences between each slot and the slot of the first burst in the sequence and transpose to a common time reference. Convert time differences to slot differences.
	9	endrep		k:= k + 1		
	10	verify		MAX(slot_diff(n)) - MIN(slot_diff(n)) ≤ V12 × M1/V11		Verify distribution of slots is over candidate slot range.
	11	record		num_slot_diff(m):= 0 for all m		Initialize the number of slots in each candidate slot position to zero.
	12	rep 110		n:= 1		
	13	record		num_slot_diff(slot_diff(n)):= num_slot_diff(slot_diff(n)) + 1		Record the frequency of occurrence of slots in each candidate slot position.
	14	endrep		n:= n + 1		
	15	rep m		m:= MIN(slot_diff(n)); chi_squared:= 0		Set initial value of m to the minimum value of slot_diff.
	16	record		chi_squared:= chi_squared + $(num\_slot\_diff(m) - 10)^2/10$		The distribution is tested for uniformity by calculating the value of chi_squared.
	17	until		m:= MAX(slot_diff(n))		
	18	verify		chi_squared < 21,2		Value of chi_squared shall be less than 21,2 for confidence that the distribution is uniform (10 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
postamble	19	send	VSS	SET PARAMETERS ($TV11_{min}$:= 4; $TV11_{max}$:= 8; V12:= 0.1) associated with sync burst generation		Reset to default values.
Comments:						

Test Case Name:	Sync_Fixed_NIC					
Purpose:	**To demonstrate that a station sets the navigation integrity category appropriately.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	rep 2		ni:= {3, 6}		Repeat for two values of NIC.
	3	send	Position	From a source with nic:= ni apply position parameters as: lat:= 0; lon:= E 21 NM		Apply position data of known NIC category to Position PCO.
	4	await	RF	SYNC_BURST_I (s= add_A)	SI	Wait for a sync burst from the station under test.
	5	verify	RF	lat = CPR_LAT(0) lon = CPR_LON(E 21 NM) nic = 3		Verify that the lat and lon data is correct and that the NIC value is appropriate to the source of position data.
	6	do	Position	Remove previously applied parameters		Remove position data from Position PCO.
	7	wait		4 s		Wait 4 s.
	8	await	RF	SYNC_BURST_I (s= add_A)	SI	Wait for a sync burst from the station under test.
	9	verify	RF	nic = 0		Verify that the NIC field indicates no position data available.
	10	endrep		next ni		Repeat for second value of NIC.
postamble	11					Bring test equipment into idle state.
Comments:						

Test Case Name:	Sync_Fixed_BaseAlt					
Purpose:	**To demonstrate that a station sets the base altitude in the fixed part of the sync burst in accordance with the input altitude data.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	rep 8		n:= {-1399, -6, 7999, 8015, 71912.5, 72400, 130049.5, 130051}; m:= {1, 131, 932, 934, 3490, 3495, 4071, 4072}		
	3	send	Altitude	Apply base altitude parameter as: altitude = n AND Apply baro/geo altitude parameter as: baro/geo = 0		Apply altitude data and baro/geo altitude parameter to Altitude PCO.
	4	await	RF	SYNC_BURST_I (s= add_A)	SI	Wait for a sync burst from the station under test.
	5	record	RF	BALT:= balt B/G:= b/g		Record the balt value.
	6	verify		BALT = m B/G:= 0		Verify that balt and b/g are correctly transmitted in the sync burst.
	7	endrep		next n		
	8	send	Altitude	Apply base altitude parameter as: altitude = station on ground AND Apply baro/geo altitude parameter as: baro/geo = 0		Apply altitude 'station on ground" and baro/geo altitude parameter to Altitude PCO.
	9	await	RF	SYNC_BURST_I (s= add_A)	SI	Wait for a sync burst from the station under test.
	10	record	RF	BALT:= balt B/G:= b/g		Record the balt value.
	11	verify		BALT = 4095 B/G:= 0		Verify that balt and b/g are correctly transmitted in the sync burst.
	12	do	Altitude	Remove previously applied altitude parameter		Remove data at altitude PCO.
	13	await	RF	SYNC_BURST_I (s= add_A)	SI	Wait for a sync burst from the station under test.
	14	record	RF	BALT:= balt		Record the balt value.
	15	verify		BALT = 0		Verify that balt = 0 is transmitted in the sync burst.
Comments:						

Test Case Name:	**Sync_Fixed_DataAge**					
Purpose:	**To demonstrate that a station sets the data age subfield of a sync burst appropriately.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	send	VSS	SET PARAMETERS (V11:= 60) associated with sync burst generation		Set the station under test to transmit bursts at the rate of 1 a second.
	3	send	Position	Apply position parameters as: lat:= 0; lon:= E 21 NM		Apply position data of known NIC category to Position PCO.
	4	await	RF	SYNC_BURST_I (s= add_A)	SI	Wait for a sync burst from the station under test.
	5	verify	RF	lat = CPR_LAT(0) lon:= CPR_LON(E 21 NM) 1 ≤ nic ≤ 11		Verify that the nic value indicates valid position data.
	6	do	Position	Remove previously applied parameters		Remove position data from Position PCO.
	7	await	RF	SYNC_BURST_I (s= add_A)	SI	Wait for the next sync burst.
	8	record	RF	DA:= da		
	9	await	RF	SYNC_BURST_I (s= add_A)	SI	Wait for the following sync burst.
	10	record	RF	DA2:= da		
	11	verify		decoded_latency(DA2) - decoded_latency(DA) = 1 000 ± 200 ms		Verify data age subfield represents 1 s (±200 ms) greater than the data age subfield in the previous sync burst.
	12	await	RF	SYNC_BURST_I (s= add_A)	SI	Wait for the following sync burst.
	13	record	RF	DA3:= da		
	14	verify		decoded_latency(DA3) - decoded_latency(DA) = 2000 ± 200 ms		Verify data age subfield represents 1 s (±200 ms) greater than the data age subfield in the previous sync burst.
	15	await	RF	SYNC_BURST_I (s= add_A)	SI	Wait for the following sync burst.
	16	record	RF	DA4:= da		
	17	verify		decoded_latency(DA4) - decoded_latency(DA) = 3 000 ± 200 ms		Verify data age subfield represents 1 s (±200 ms) greater than the data age subfield in the previous sync burst.
postamble	18	send	VSS	SET PARAMETERS (V11:= 1) associated with sync burst generation		Reset to default values.
Comments:						

Test Case Name:		**NetEntry_Periodic**				
Purpose:		**To demonstrate that a station which desires to gain entry to a network using the combined periodic and incremental broadcast protocols is able to set up a series of regularly spaced streams.**				
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		switch on VDL4 transceiver		
	2	verify	Selftest	successful VDL4 transceiver selftest		Verify that the VDL4 transceiver passes power-up self-test.
	3	do		SET NETWORK ENTRY BY PERIODIC AND INCREMENTAL		Ensure transceiver is set to perform network entry by a combination of periodic and incremental broadcasts as opposed to other means.
test body	4	rep 10		n:= 1		Repeat the test n times.
	5	do		switch off VDL4 transceiver		
	6	wait		15 s		Ensure network entry will be triggered by waiting a sufficient time.
	7	do		switch on VDL4 transceiver		
	8	verify	Selftest	successful VDL4 transceiver selftest		Verify that the VDL4 transceiver passes power-up self-test.
	9	record		t:= time at beginning of first slot at which transceiver is able to receive incoming transmissions		
	10	verify	RF	No transmissions from the station under test before time:= t + 61.68		Ensure there are no transmissions from the station under test for a period of M1 + 126 slots after start up, in which time the station shall be listening to the channel to build up a complete slot map.
	11	await	RF	SYNC_BURST_c (s= add_A) transmitted at or after time:= t + 61.68	Sc	Verify an autonomous sync burst is then transmitted.
	12	record	RF	sync_time:= time at beginning of slot occupied by SYNC_BURST_c (s= add_A)	Sc	
	13	verify	RF	SYNC_BURST_c (s= add_A) contains pt = 3 AND io ≠ 0 (or po ≠ 0)	Sc	Verify that the first sync burst transmitted contains pt and io (or po) values compatible with a combined periodic and incremental broadcast reservation.
	14	record	RF	IO:= io contained in SYNC_BURST_c (s= add_A)	Sc	
	15	await		time:= sync_time + IO × 4 × 60/M1		
	16	verify	RF	SYNC_BURST_c (s= add_A) contained in slot at time:= sync_time + IO × 4 × 60/M1	Sc	Verify that a further sync burst is made in the slot identified by the io value contained in the first sync burst.
	17	await		time:= sync_time + 60		
	18	verify	RF	SYNC_BURST_c (s= add_A) contained in slot at time:= sync_time + 60	Sc	Verify that a sync burst is contained in the slot that occurs one superframe after the first sync burst.
	19	verify	RF	IF SYNC_BURST_c (s= add_A) in slot at time:= sync_time + 60 contains pt = 3 THEN po = 0	Sc	Verify that if this sync burst contains pt = 3 that it also contains po = 0.
	20	endrep		n:= n + 1		
postamble	21					Bring test equipment into idle state.
Comments: This test requires waiting for one minute to perform net entry. If this is not supported by a station, then this test does not apply.						

Test Case Name:	NetEntry_Delayed_Plea					
Purpose:	**To demonstrate that a station which desires to perform network entry using a delayed plea transmission will make such a transmission in an otherwise unoccupied slot.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		switch on VDL4 transceiver		
	2	verify	Selftest	successful VDL4 transceiver selftest		Verify that the VDL4 transceiver passes power-up self-test.
	3	do		SET NETWORK ENTRY BY DELAYED PLEA		Ensure transceiver is set to perform network entry by delayed plea transmission as opposed to other means.
test body	4	rep 10		n:= 1		Repeat the test n times.
	5	do		switch off VDL4 transceiver		
	6	wait		15 s		Ensure network entry will be triggered by waiting a sufficient time.
	7	do		switch on VDL4 transceiver		
	8	verify	Selftest	successful VDL4 transceiver selftest		Verify that the VDL4 transceiver passes power-up self-test.
	9	record		t:= time at beginning of first slot at which the station under test becomes operational		
	10	rep 45		m:= 1; plea_in_unoccupied:= FALSE; plea_in_occupied:= FALSE		Over 1 superframe, fill 99 % of slots with random access transmissions or sync bursts from a simulated station B.
	11	rep 99		p:= 1		
	12	send	RF	SYNC_BURST_b (po:= 0; pt:= 0; s= add_B) in slot beginning at time = sync_time + (m - 1) × 100 + p	Sb	Send a sync burst from station B in each slot.
	13	 record record verify	 RF RF	IF PLEA_a (s:= add_A) is present in any part of slot beginning at sync_time + (m - 1) × 100 + p THEN plea_in_occupied:= TRUE ELSE SYNC_BURST_b (po:= 0; pt:= 0; s= add_B) is only transmission present in slot beginning at sync_time + (m - 1) × 100 + p	Pa, Sb	Record whether a plea transmission is made by the station under test in a slot already occupied by a sync burst.
	14	endrep		p = 99; p:= p + 1		
	15	 record record verify	 RF RF	IF PLEA_a (s:= add_A) present in slot beginning at sync_time + (m - 1) × 100 + 100 THEN plea_in_unoccupied:= TRUE ELSE no transmission present in slot beginning at sync_time + (m - 1) × 100 + 100	Pa	Record whether a plea transmission is made by the station under test in an unoccupied slot.
	16	endrep		m:= m + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	17	verify		plea_in_unoccupied:= TRUE AND plea_in_occupied:= FALSE		After 1 superframe, verify that the station under test has made at least one plea transmission in one of the few otherwise unoccupied slots, and has made no plea transmissions in otherwise occupied slots.
	18	endrep		n:= n + 1		
postamble	19					Bring test equipment into idle state.
Comments: Network entry by delayed plea transmission is not mandated by ICAO standards, and nor by the present document. Step 3 is provided to ensure that this means of net entry is selected in preference to other means, such as waiting for one minute. In the event that the transceiver under test does not support network entry by delayed plea transmissions, then this test does not apply.						

Test Case Name:	NetEntry_Delayed_BND					
Purpose:	**To demonstrate that a station which desires to perform network entry using a delayed BND transmission will make such a transmission in an otherwise unoccupied slot.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		switch on VDL4 transceiver		
	2	verify	Selftest	successful VDL4 transceiver selftest		Verify that the VDL4 transceiver passes power-up self-test.
	3	do		SET NETWORK ENTRY BY DELAYED BND		Ensure transceiver is set to perform network entry by delayed BND transmission as opposed to other means.
test body	4	rep 10		n:= 1		Repeat the test n times.
	5	do		switch off VDL4 transceiver		
	6	wait		15 s		Ensure network entry will be triggered by waiting a sufficient time.
	7	do		switch on VDL4 transceiver		
	8	verify	Selftest	successful VDL4 transceiver selftest		Verify that the VDL4 transceiver passes power-up self-test.
	9	record		t:= time at beginning of first slot at which the station under test becomes operational		Send a sync burst from station B.
	10	rep 45		m:= 1; bnd_in_unoccupied:= FALSE; bnd_in_occupied:= FALSE		Over 1 superframe, fill 99 % of slots with random access transmissions or sync bursts from a simulated station B.
	11	rep 99		p:= 1		
	12	send	RF	SYNC_BURST_b (po:= 0; pt:= 0; s= add_B) in slot beginning at time = sync_time + (m - 1) × 100 + p	Sb	Send a sync burst from station B in each slot.
	13	 record record verify	 RF RF	IF BND_DELAYED_a (s:= add_A) is present in any part of slot beginning at sync_time + (m - 1) × 100 + p THEN bnd_in_occupied:= TRUE ELSE SYNC_BURST_b (po:= 0; pt:= 0; s= add_B) is only transmission present in slot beginning at sync_time + (m - 1) × 100 + p	BDa, Sb	Record whether a BND transmission is made by the station under test in a slot already occupied by a sync burst.
	14	endrep		p = 99; p:= p + 1		
	15	 record record verify	 RF RF	IF BND_DELAYED_a (s:= add_A) present in slot beginning at sync_time + (m - 1) × 100 + 100 THEN bnd_in_unoccupied:= TRUE ELSE no transmission present in slot beginning at sync_time + (m - 1) × 100 + 100	BDa	Record whether a BND transmission is made by the station under test in an unoccupied slot.
	16	endrep		m:= m + 1		

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	17	verify		bnd_in_unoccupied:= TRUE AND bnd_in_occupied:= FALSE		After 1 superframe, verify that the station under test has made at least one BND transmission in one of the few otherwise unoccupied slots, and has made no BND transmissions in otherwise occupied slots.
	18	endrep		n:= n + 1		
postamble	19					Bring test equipment into idle state.
Comments: Network entry by delayed BND transmission is not mandated by ICAO standards, and nor by the present document. Step 3 is provided to ensure that this means of net entry is selected in preference to other means, such as waiting for one minute. In the event that the transceiver under test does not support network entry by delayed BND transmissions, then this test does not apply.						

Test Case Name:	**NetEntry_Receive**					
Purpose:	**To demonstrate that a station in receipt of a delayed transmission containing a plea will generate a reply to the source station with slots for it to transmit in, if it has some slots which it could make available.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b (V11:= 10)	Sb	Set up a series of periodic streams of one-slot messages from the station under test. V11 set to 10 bursts within M1 slots.
	3	await	RF	SYNC_BURST_b (s= add_A)	Sb	Wait for the first sync burst to be transmitted by the station under test.
	4	send	RF	PLEA_a (s= add_B; d= add_A)	Pa	Send a delayed plea transmission from a simulated station B to the station under test.
	5	record	RF	plea_time:= time at beginning of slot containing PLEA_a (s= add_B; d= add_A)	Pa	
	6	verify	RF	PLEA_RESP_a (s= add_A; d= add_B) with $a_1 \neq 0$ OR PLEA_RESP_b (s= add_A; d= add_B) with $a_1 \neq 0$ transmitted before time:= plea_time + 2	PRa, PRb	Verify that a plea response is issued by the station under test addressed to station B within TL5 seconds and that it contains at least one slot position (in a_1) for station B to use for transmission.
postamble	7	send	VSS	SET PARAMETERS (V11:= 1)		Reset to default values.
Comments:						

Test Case Name:	**NetEntry_OneMinute**					
Purpose:	**To demonstrate that a station which desires to transmit for the first time without using network entry protocols, will listen to the channel on which it desires to transmit for M1 + 126 slots prior to making any transmissions.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		switch on VDL4 transceiver		
	2	verify	Selftest	successful VDL4 transceiver selftest		Verify that the VDL4 transceiver passes power-up self-test.
	3	do		SET NETWORK ENTRY BY WAITING ONE MINUTE		Ensure transceiver is set to perform network entry by waiting for one minute as opposed to other means.
test body	4	rep 10		n:= 1		Repeat the test n times.
	5	do		switch off VDL4 transceiver		
	6	wait		15 s		Ensure network entry will be triggered by waiting a sufficient time.
	7	do		switch on VDL4 transceiver		
	8	verify	Selftest	successful VDL4 transceiver selftest		Verify that the VDL4 transceiver passes power-up self-test.
	9	record		t:= time at beginning of first slot at which transceiver is able to receive incoming transmissions		
	10	verify	RF	No transmissions from the station under test before time:= t + 61.68		Ensure there are no transmissions from the station under test for a period of M1 + 126 slots after start up, in which time the station shall be listening to the channel to build up a complete slot map.
	11	verify	RF	SYNC_BURST_c (s= add_A) transmitted at or after time:= t + 61.68	Sc	Verify an autonomous sync burst is then transmitted.
	12	endrep		n:= n + 1		
postamble	13					Bring test equipment into idle state.
Comments: Network entry by waiting one minute is not mandated by ICAO standards, and nor by the present document. Step 3 is provided to ensure that this means of net entry is selected in preference to other means, such as a delayed plea transmission. In the event that the transceiver under test does not support network entry by waiting one minute, then this test does not apply.						

Test Case Name:	**NetEntry_Repeat**					
Purpose:	**To demonstrate that a station does not repeat net entry within M1 slots following the previous successful net entry.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS (V11:= 60)		Set the station under test to transmit 1 report per second.
	3	wait		1 minute		
test body	4	await	RF	SYNC_BURST_c (s= add_A)	Sc	Verify an autonomous sync burst is transmitted, indicating that the station is already on the network.
	5	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 1; CG1_decay:= 1/247; CG1_inc:= 1; L1:= 1)		Set CG1_reach to 1 minute, CG1_limit to 1, CG1_decay to 1/247, CG1_inc to 1 and L1 to 1, while leaving the other parameters at their default values.
	6	await	RF	SYNC_BURST_c (s= add_A)	Sc	Wait for the first sync burst after the UCTRL.
	7	record	RF	sync_time:= time at beginning of slot occupied by SYNC_BURST_c	Sc	
	8	await		time = sync_time + 40		
	9	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM)) (position of station B is < CG1_range away from station under test) in slot beginning at time = sync_time + 40	Sa	Send a sync burst from a simulated station B < CG1_range away from the station under test.
	10	verify	RF	SYNC_BURST_c (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= sync_time + 60	Sc	Verify that network entry has not been triggered by observing that for a period of one minute: an autonomous transmission is made by the station under test; and a BND has not been transmitted; and a plea has not been transmitted.
	11	await		time = sync_time + 130		
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_C; lat:= CPR_LAT(0); lon:= CPR_LON(E 80 NM)) (position of station C is < CG1_range away from station under test) in slot beginning at time = sync_time + 130	Sa	Send a sync burst from a simulated station C < CG1_range away from the station under test.
	13	verify	RF	Before time:= sync_time + 131.68 EITHER No transmissions OR Plea transmitted in delayed burst OR BND transmitted in delayed burst		Verify that network entry has been triggered by observing that for a period of one minute: no transmissions are made by the station under test; or a BND has been transmitted; or a plea has been transmitted.
	14	await	RF	SYNC_BURST_c (s= add_A) transmitted at or after time:= sync_time + 131.68	Sc	Verify an autonomous sync burst is then transmitted.
	15	record	RF	sync_time2:= time at beginning of slot occupied by SYNC_BURST_c	Sc	
	16	await		time = sync_time2 + 20		
	17	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 50 NM))	Sa	Send a sync burst from a simulated station D < CG1_range away from the station under test.

Test Case Name:	NetEntry_Repeat					
Purpose:	**To demonstrate that a station does not repeat net entry within M1 slots following the previous successful net entry.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
				(position of station D is < CG1_range away from station under test) in slot beginning at time = sync_time2 + 20		
	18	await		time = sync_time2 + 22		
	19	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_E; lat:= CPR_LAT(0); lon:= CPR_LON(E 55 NM)) (position of station E is < CG1_range away from station under test) in slot beginning at time = sync_time2 + 22	Sa	Send a sync burst from a simulated station E < CG1_range away from the station under test.
	20	verify	RF	SYNC_BURST_c (s= add_A) transmitted before time:= sync_time2 + 60	Sc	Verify that network entry has not been triggered within M1 slots following the previous successful net entry.
postamble	21	send	VSS	SET PARAMETERS (CG1_reach:= 3; CG1_limit:= 2 000; CG1_decay:= 247/256; CG1_inc:= 25; L1:= 3; V11:= 1)		Set the parameters to their default values.
Comments:						

Test Case Name:	**UCTRL_param_VS1**					
Purpose:	**To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter VS1.**					
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.
test body	3	send	RF	UCTRL_VS (VS1:= 6; VS2, VS4, VS5:= default values; s:= add_G)	Cv	Send a broadcast UCTRL from a simulated ground station G addressed to the station under test. The transmission sets the VS1 parameter to 6.
	4	send	RF	SYNC_BURST_b (pt:= 3; po:= 0; a/d:= 0; s:= add_G, address indicating source is a ground station)	Sb	Send a sync burst (burst length 1) from ground station G reserving the same transmission slot, and the following VS1=6 ground quarantined slots, in the next 4 superframes.
	5	record	RF	periodic_start:= time at beginning of slot containing SYNC_BURST_b	Sb	Provides a reference time for the reserved slots of the ground station G.
	6	macro		M_RAND_ACC_SU (sf:= 2)		Queue random access transmissions over 2 superframes.
	7	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	8	await		time = periodic_start + 60		
	9	rep M1		n:= 0		Verify over 1 superframe.
	10	 verify verify	 RF RF	IF n = {0, 1, 2, 3, 4, 5, 6} THEN no transmission present in slot beginning at time = periodic_start + (n + M1) × 60/M1 ELSE RAND_ACC_DATA_a (s= add_A) in slot beginning at time = periodic_start + (n + M1) × 60/M1	Ra	Verify that random access transmissions are made by the station under test in all slots except the reserved slots and the following ground quarantined slots.
	11	endrep		n:= n + 1		
postamble	12	send	VSS	SET PARAMETERS (p:= 64/256)		Reset to default value.
	13	send	RF	UCTRL_VS (VS1, VS2, VS4, VS5:= default values; s:= add_G)	Cv	Send a broadcast UCTRL from ground station G addressed to the station under test, setting the VS1 parameter back to its default value.
	14	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**UCTRL_param_VS2**					
Purpose:	**To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter VS2.**					
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.
	3	send	VSS	SET PARAMETERS (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
test body	4	send	RF	UCTRL_VS (VS2:= 14; VS1, VS4, VS5:= default values; s:= add_G)	Cv	Send a broadcast UCTRL from a simulated ground station G addressed to the station under test. The transmission sets the VS2 parameter to 14, representing a CCI ratio of 5 instead of the default value of 4.
	5	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	6	await	RF	INCREM_BURST_a (s= add_A)	Ia	Wait for the incremental broadcast reservation.
	7	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s= add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	8	record	RF	IO(0):= io contained in INCREM_BURST _a (s= add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	9	record		random_position:= 64 + 4 × RAND(0, 5)		Slot to reserve within each candidate range, chosen at random from the six possible candidate slots.
	10	record		no_IO(m):= 0 for m:= {64, 68, 72, 76, 80, 84}		Initialize the number of slots in each candidate slot position to zero.
	11	rep 60		n:= 1		Repeat 60 times.
	12	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 160 NM)) (position of station B is > Q2a away from station under test) in slot beginning at time = current_inc_time + 5 × 60/M1	Sa	Send a sync burst from a simulated station B > Q2a away from the station under test, reporting B's position.
	13	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 195 NM)) (position of station D is such that a transmission from B to D is not CCI protected with a CCI ratio of 5 but would be with a CCI ratio of 4) in slot beginning at time = current_inc_time + 10 × 60/M1	Sa	Send a sync burst from a simulated station D, reporting D's position, which is such that a transmission from B to D is not CCI protected with a CCI ratio of 5 (but would be with a CCI ratio of 4).
	14	record		reserve_slot:= 4 × IO(n - 1) + random_position		Slot position to reserve within the next-but-one incremental broadcast candidate range.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	15	send	RF	UNI_BURST_a (sdf:= 1; ro:= reserve_slot - 15 - 1; lg:= 0; pr:= 0; s:= add_B; d:= add_D) in slot beginning at time = current_inc_time + 15 × 60/M1	Ua	Send a unicast burst from station B > Q2a away from A, reserving a slot for transmission to station D. The distance from the station under test (station A) to station D is such that a transmission from B to D is not CCI protected with a CCI ratio of 5 (but would be with a CCI ratio of 4). The burst reserves a slot in the candidate range of the next-but-one incremental broadcast reservation.
	16	await	RF	INCREM_BURST_a (s= add_A)	Ia	Wait for the next incremental broadcast reservation.
	17	record	RF	current_inc_time:= time at beginning of slot containing INCREM_BURST_a (s= add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	18	record	RF	IO(n):= io contained in INCREM_BURST _a (s= add_A) no_IO(IO(n)):= no_IO(IO(n)) + 1	Ia	Record value of io given in the incremental broadcast reservation. Record the frequency of occurrence of slots in each candidate slot position.
	19	endrep		n:= n + 1		
	20	verify		no_IO(random_position) = 0		Verify that no transmission is made in the slot reserved by station B.
	21	rep 6		m:= 64; chi_squared:= 0		Set value of m to the minimum value of the candidate range. Initialize chi_squared.
	22	 record		IF m ≠ random_position THEN chi_squared:= chi_squared + $(no\_IO(m) - 10)^2/10$		For all the other slots the distribution is tested for uniformity by calculating the value of chi_squared.
	23	endrep		m:= m + 4		
	24	verify		chi_squared < 11,7		Value of chi_squared shall be less than 11,7 for confidence that the distribution is uniform (4 degrees of freedom). The test should be repeated if the value of chi_squared exceeds this value (this will normally happen with a uniform distribution on only 2 % of occasions).
postamble	25	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18; VS2:= 12)		Reset to default values.
	26	send	RF	UCTRL_VS (VS1, VS2, VS4, VS5:= default values; s:= add_G)	Cv	Send a broadcast UCTRL from ground station G addressed to the station under test, setting the VS2 parameter back to its default value.
	27	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**UCTRL_param_VS4**					
Purpose:	**To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter VS4.**					
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.
test body	3	send	RF	UCTRL_VS (VS4:= 5; VS1, VS2, VS5:= default values; s:= add_G)	Cv	Send a broadcast UCTRL from a simulated ground station G addressed to the station under test. The transmission sets the VS4 parameter to 50 NM.
	4	send	RF	SYNC_BURST_k(3) (pt:= 3; po:= 0; a/d:= 1; s:= add_G, address indicating source is a ground station; lat:= CPR_LAT(0); lon:= CPR_LON(E 100 NM)) (position of ground station G is > VS4 and < Q2c away from station under test)	Sk(3)	Send a sync burst 3 slots in length from ground station G > VS4 and < Q2c away from the station under test, reporting the ground station's position.
	5	record	RF	start_time:= time at beginning of slot containing SYNC_BURST_k(3)	Sk(3)	
	6	macro		M_RAND_ACC_SU (sf:= 4)		Queue random access transmissions over 4 superframes.
	7	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions.
	8	rep 4×M1		p:= 0		
	9	 verify verify	 RF RF	IF p = {0, 1, 2, 4500, 4501, 4502, 9000, 9001, 9002, 13500, 13501, 13502} THEN no transmission present in slot beginning at time = start_time + p × 60/M1 ELSE RAND_ACC_DATA_a (s= add_A) in slot beginning at time = start_time + p × 60/M1	Ra	Verify that random access transmissions are not made by the station under test in the slots reserved by the ground station, but are made in the slots immediately following the reserved slots.
	10	endrep		p:= p + 1		
postamble	11	send	RF	UCTRL_VS (VS1, VS2, VS4, VS5:= default values; s:= add_G)	CXe	Send a broadcast UCTRL from ground station G addressed to the station under test, setting the VS4 parameter to its default value.
	12	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**UCTRL_param_Q4**					
Purpose:	**To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of private parameter Q4.**					
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 (Q4:= 6; V22:= 12)		Q4 set to 6; equals the number of slots in the incremental broadcast dither range available for selection. V21 (nominal incremental reserved slot position) equals default value of 75 slots. V22 (max incremental dither range) set to 12; gives maximum dither range of 75 ± 12 after the incremental broadcast transmission slot (allowed slots of 64, 68, 72, 76, 80, 84).
test body	4	send	VSS	INCREMENTAL BROADCAST request to transmit INCREM_BURST_a followed by successive INCREM_BURST_a in reserved slots	Ia	Request to send incremental broadcast reservation and to place another incremental broadcast reservation in each reserved slot, thus creating an automatic succession of incremental broadcast reservations.
	5	await	RF	INCREM_BURST_a (s= add_A)	Ia	Wait for the first incremental broadcast reservation from the station under test (incremental burst 1).
	6	record	RF	inc_time:= time at beginning of slot containing INCREM_BURST_a (s= add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	7	record	RF	IO:= io contained in INCREM_BURST _a (s= add_A)	Ia	Record value of io given in the incremental broadcast reservation.
	8	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 110 NM)) (position of station B is < Q2a, b, c, d away from station under test) in slot beginning at time = inc_time + 5 × 60/M1	Sa	Send a sync burst from a simulated station B < Q2a, b, c, d away from the station under test, reporting B's position.
	9	send	RF	INCREM_BURST_b(16) (io:= 24; s= add_B) in slot beginning at time = inc_time + (4 × IO + 70 - 96) × 60/M1	Ib(16)	Send an incremental burst from station B < Q2a, b, c, d away reserving a series of 16 slots that conflict with all but two slots of the candidate range of the next incremental burst from the station under test.
	10	send	RF	UCTRL_Q (Q4:= 2; Q1min, Q1max, Q2a, Q2b, Q2c, Q2d: default values; s:= add_G; d:= add_A)	Cq	Send a broadcast UCTRL from a simulated ground station G addressed to the station under test. The transmission sets the Q4 parameter to 2.
	11	verify	RF	INCREM_BURST_a (s= add_A) in slot beginning at time = inc_time + (4 × IO) × 60/M1	Ia	Verify that the station under test uses the fact that Q4 has been reduced to 2 by being able to select a slot from the two available slots not reserved by station B, when it would not be able to do so with Q4 set to 3. This slot therefore contains an incremental broadcast reservation (incremental burst 2) pointing to the selected slot.
	12	record	RF	inc_time:= time at beginning of slot containing INCREM_BURST_a (s= add_A)	Ia	Record the time of the incremental reservation transmission slot as current_inc_time.
	13	record	RF	IO2:= io contained in INCREM_BURST _a (s= add_A)	Ia	Record value of io given in the incremental broadcast reservation.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	verify	RF	INCREM_BURST_a (s= add_A) in slot beginning at time = inc_time_2 + (4 × IO2) × 60/M1	Ia	Verify that the selected slot is used by the station under test to transmit a further incremental broadcast (incremental burst 3).
Postamble	15	send	VSS	SET PARAMETERS (Q4:= 3; V22:= 18)		Reset to default value.
	16	send	RF	UCTRL_q (Q4, Q1min, Q1max, Q2a, Q2b, Q2c, Q2d: default values; s:= add_G)	Cq	Send a broadcast UCTRL from ground station G addressed to the station under test, setting the Q4 parameter to its default value.
	17	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	UCTRL_param_CG1_limit					
Purpose:	**To demonstrate that a station will act correctly upon modification of parameter CG1_limit.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	await	RF	SYNC_BURST_c (s= add_A)	Sc	Verify an autonomous sync burst is transmitted, indicating that the station is already on the network.
	3	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 260; CG1_decay:= 1; CG1_inc:= 256; L1:= 1)		Set CG1_reach to 1 minute, CG1_limit to 260, CG1_decay to its maximum value (minimum rate of decay), CG1_inc to maximum and L1 to 1, while leaving the other parameters at their default values.
	4	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 90 NM)) (position of station B is < CG1_range away from station under test)	Sa	Send a sync burst from a simulated station B < CG1_range away from the station under test.
	5	record	RF	sync_time:= time at beginning of slot occupied by SYNC_BURST_a (s= add_B)	Sa	
	6	verify	RF	SYNC_BURST_c (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= sync_time + 60	Sc	Verify that network entry has not been triggered by observing that for a period of one minute: an autonomous transmission is made by the station under test; and a BND has not been transmitted; and a plea has not been transmitted.
	7	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 250; CG1_decay:= 1; CG1_inc:= 256; L1:= 1)		Set CG1_reach to 1 minute, CG1_limit to 250, CG1_decay to its maximum value (minimum rate of decay), CG1_inc to maximum and L1 to 1, while leaving the other parameters at their default values.
	8	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 90 NM)) (position of station B is < CG1_range away from station under test)	Sa	Send a sync burst from a simulated station B < CG1_range away from the station under test.
	9	record	RF	net_time:= time at beginning of slot occupied by SYNC_BURST_a (s= add_B)	Sa	
	10	verify	RF	Before time:= net_time + 60 + 61.68 EITHER No transmissions OR Plea transmitted in delayed burst OR BND transmitted in delayed burst		Verify that network entry has been triggered by observing that for a period of M1 + 126 slots: no transmissions are made by the station under test; or a BND has been transmitted; or a plea has been transmitted.
postamble	11	send	VSS	SET PARAMETERS (CG1_reach:= 3; CG1_limit:= 2 000; CG1_decay:= 247/256; CG1_inc:= 25; L1:= 3)		Set the parameters to their default values.
Comments:						

Test Case Name:	UCTRL_param_CG1_range					
Purpose:	**To demonstrate that a station will act correctly upon modification of private parameter CG1_range.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	await	RF	SYNC_BURST_c (s= add_A)	Sc	Verify an autonomous sync burst is transmitted, indicating that the station is already on the network.
	3	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 1; CG1_decay:= 1; CG1_inc:= 256; CG1_range:= 70; L1:= 1)		Set CG1_reach to 1 minute, CG1_limit to 1, CG1_decay to its maximum value (minimum rate of decay), CG1_inc to maximum, CG1_range to 70 NM and L1 to 1, while leaving the other parameters at their default values.
	4	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 80 NM)) (position of station B is > CG1_range away from station under test)	Sa	Send a sync burst from a simulated station B > CG1_range away from the station under test.
	5	record	RF	sync_time:= time at beginning of slot occupied by SYNC_BURST_a (s= add_A)	Sa	
	6	verify	RF	SYNC_BURST_c (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= sync_time + 60	Sc	Verify that network entry has not been triggered by observing that for a period of one minute: an autonomous transmission is made by the station under test; and a BND has not been transmitted; and a plea has not been transmitted.
	7	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 1; CG1_decay:= 1; CG1_inc:= 256; CG1_range:= 90; L1:= 1)		Set CG1_reach to 1 minute, CG1_limit to 1, CG1_decay to its maximum value (minimum rate of decay), CG1_inc to maximum, CG1_range to 90 NM and L1 to 1, while leaving the other parameters at their default values.
	8	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 80 NM)) (position of station B is < CG1_range away from station under test)	Sa	Send a sync burst from station B that is now < CG1_range away from the station under test.
	9	record	RF	net_time:= time at beginning of slot occupied by SYNC_BURST_a (s= add_B)	Sa	
	10	verify	RF	Before time:= net_time + 60 + 61.68 EITHER No transmissions OR Plea transmitted in delayed burst OR BND transmitted in delayed burst		Verify that network entry has been triggered by observing that for a period of M1 + 126 slots: no transmissions are made by the station under test; or a BND has been transmitted; or a plea has been transmitted.
postamble	11	send	VSS	SET PARAMETERS (CG1_reach:= 3; CG1_limit:= 2 000; CG1_decay:= 247/256; CG1_inc:= 25; L1:= 3)		Set the parameters to their default values.
Comments:						

Test Case Name:	UCTRL_param_CG1_inc					
Purpose:	**To demonstrate that a station will act correctly upon modification of private parameter CG1_inc.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	await	RF	SYNC_BURST_c (s= add_A)	Sc	Verify an autonomous sync burst is transmitted, indicating that the station is already on the network.
	3	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 5; CG1_decay:= 1; CG1_inc:= 4; L1:= 1)		Set CG1_reach to 1 minute, CG1_limit to 5, CG1_decay to its maximum value (minimum rate of decay), CG1_inc to 4 and L1 to 1, while leaving the other parameters at their default values.
	4	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM)) (position of station B is < CG1_range away from station under test)	Sa	Send a sync burst from a simulated station B < CG1_range away from the station under test.
	5	record	RF	sync_time:= time at beginning of slot occupied by SYNC_BURST_a (s= add_A)		
	6	verify	RF	SYNC_BURST_c (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= sync_time + 60	Sc	Verify that network entry has not been triggered by observing that for a period of one minute: an autonomous transmission is made by the station under test; and a BND has not been transmitted; and a plea has not been transmitted.
	7	send	RF	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 5; CG1_decay:= 1; CG1_inc:= 6; L1:= 1)		Set CG1_reach to 1 minute, CG1_limit to 5, CG1_decay to its maximum value (minimum rate of decay), CG1_inc to 6 and L1 to 1, while leaving the other parameters at their default values.
	8	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_H; lat:= CPR_LAT(0); lon:= CPR_LON(E 85 NM)) (position of station H is < CG1_range away from station under test)	Sa	Send a sync burst from a simulated station H < CG1_range away from the station under test.
	9	record	RF	net_time:= time at beginning of slot occupied by SYNC_BURST_a (s= add_A)	Sa	
	10	verify	RF	Before time:= net_time + 60 + 61.68 EITHER No transmissions OR Plea transmitted in delayed burst OR BND transmitted in delayed burst		Verify that network entry has been triggered by observing that for a period of M1 + 126 slots: no transmissions are made by the station under test; or a BND has been transmitted; or a plea has been transmitted.
postamble	11	send	RF	SET PARAMETERS (CG1_reach:= 3; CG1_limit:= 2 000; CG1_decay:= 247/256; CG1_inc:= 25; L1:= 3)		Set the parameters to their default values.
Comments:						

Test Case Name:	**Param_CG1_reach**					
Purpose:	**To demonstrate that a station will act correctly upon modification of private parameter CG1_reach.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	await	RF	SYNC_BURST_c (s= add_A)	Sc	Verify an autonomous sync burst is transmitted, indicating that the station is already on the network.
	3	send	VSS	SET PARAMETERS (CG1_reach:= 2; CG1_limit:= 5; CG1_decay:= 1; CG1_inc:= 1; L1:= 1)		Set CG1_reach to 2 minutes, CG1_limit to 5, CG1_decay to its maximum value (minimum rate of decay), and CG1_inc to 1, while leaving the other parameters at their default values. Set L1 to 1.
	4	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM)) (position of station B is < CG1_range away from station under test)	Sa	Send a sync burst from a simulated station B < CG1_range away from the station under test. The purpose of sending this sync burst is so that station B will not have been unreachable for > CG1_reach minutes when the next sync burst is received.
	5	record	RF	sync_time1:= time at beginning of slot occupied by SYNC_BURST_a	Sa	
	6	verify	RF	SYNC_BURST_c (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= sync_time1 + 60	Sc	Verify that network entry has not been triggered by observing that for a period of one minute: an autonomous transmission is made by the station under test; and a BND has not been transmitted; and a plea has not been transmitted.
	7	send	VSS	SET PARAMETERS (CG1_reach:= 2; CG1_limit:= 5; CG1_decay:= 1; CG1_inc:= 10; L1:= 1)		Set CG1_reach to 2 minutes, CG1_limit to 5, CG1_decay to its maximum value (minimum rate of decay), and CG1_inc to 10, while leaving the other parameters at their default values. Set L1 to 1.
	8	await		time = sync_time1 + 170		Wait 170 s since the last sync burst from station B.
	9	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM)) (position of station B is < CG1_range away from station under test) in slot beginning at time = sync_time1 + 170	Sa	Send a sync burst from station B < CG1_range away from the station under test.
	10	record	RF	sync_time2:= time at beginning of slot occupied by SYNC_BURST_a	Sa	
	11	verify	RF	SYNC_BURST_c (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= sync_time2 + 60	Sc	Verify that network entry has not been triggered by observing that for a period of one minute: an autonomous transmission is made by the station under test; and a BND has not been transmitted; and a plea has not been transmitted.

Test Case Name:	Param_CG1_reach					
Purpose:	**To demonstrate that a station will act correctly upon modification of private parameter CG1_reach.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	12	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 5; CG1_decay:= 1; CG1_inc:= 10; L1:= 1)		Set CG1_reach to 1 minute, CG1_limit to 5, CG1_decay to its maximum value (minimum rate of decay), and CG1_inc to 10, while leaving the other parameters at their default values. Set L1 to 1.
	13	await		time = sync_time2 + 170		Wait 170 s since last sync burst from station B.
	14	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM)) (position of station B is < CG1_range away from station under test) in slot beginning at time = sync_time2 + 170	Sa	Send a sync burst from station B < CG1_range away from the station under test.
	15	verify	RF	Before time:= sync_time2 + 170 + 61.68 EITHER No transmissions OR Plea transmitted in delayed burst OR BND transmitted in delayed burst		Verify that network entry has been triggered by observing that for a period of M1 + 126 slots either: no transmissions are made by the station under test; or a BND has been transmitted; or a plea has been transmitted.
postamble	16	send	VSS	SET PARAMETERS (CG1_reach:= 3; CG1_limit:= 2 000; CG1_decay:= 247/256; CG1_inc:= 25; L1:= 3)		Set the parameters to their default values.
Comments:						

Test Case Name:	UCTRL_param_CG1_decay					
Purpose:	**To demonstrate that a station will act correctly upon modification of private parameter CG1_decay.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	await	RF	SYNC_BURST_c (s= add_A)	Sc	Verify an autonomous sync burst is transmitted, indicating that the station is already on the network.
	3	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 13; CG1_inc:= 8; CG1_decay:= ½; L1:= 1)		Set CG1_reach to 1 minute, CG1_limit to 13, CG1_inc to 8, CG1_decay to ½ and L1 to 1, while leaving the other parameters at their default values.
	4	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM)) (position of station B is < CG1_range away from station under test)	Sa	Send a sync burst from a simulated station B < CG1_range away from the station under test.
	5	record	RF	sync_time:= time at beginning of slot occupied by SYNC_BURST_a	Sa	
	6	await		time = sync_time + 1		Wait 1 s.
	7	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_C; lat:= CPR_LAT(0); lon:= CPR_LON(E 75 NM)) (position of station C is < CG1_range away from station under test) in slot beginning at time = sync_time + 1	Sa	Send a sync burst from a simulated station C < CG1_range away from the station under test.
	8	verify	RF	SYNC_BURST_c (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= sync_time + 1 + 60	Sc	Verify that network entry has not been triggered by observing that for a period of one minute: an autonomous transmission is made by the station under test; and a BND has not been transmitted; and a plea has not been transmitted.
	9	wait		3 minutes		
	10	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 13; CG1_inc:= 8; CG1_decay:= ¾; L1:= 1)		Set CG1_reach to 1 minute, CG1_limit to 13, CG1_inc to 8, CG1_decay to ¾ and L1 to 1, while leaving the other parameters at their default values.
	11	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_D; lat:= CPR_LAT(0); lon:= CPR_LON(E 85 NM)) (position of station D is < CG1_range away from station under test)	Sa	Send a sync burst from a simulated station D < CG1_range away from the station under test.
	12	record	RF	sync_time:= time at beginning of slot occupied by SYNC_BURST_a	Sa	
	13	await		time = sync_time + 1		Wait 1 s.
	14	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_F; lat:= CPR_LAT(0); lon:= CPR_LON(E 90 NM)) (position of station E is < CG1_range away from station under test)in slot beginning at time = sync_time + 1	Sa	Send a sync burst from a simulated station E < CG1_range away from the station under test.
	15	verify	RF	Before time:= sync_time + 1 + 61.68 EITHER No transmissions		Verify that network entry has been triggered by observing that for a period of M1 + 126 slots: no transmissions are made by the station under test; or

Test Case Name:	UCTRL_param_CG1_decay					
Purpose:	**To demonstrate that a station will act correctly upon modification of private parameter CG1_decay.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
				OR Plea transmitted in delayed burst OR BND transmitted in delayed burst		a BND has been transmitted; or a plea has been transmitted.
postamble	16	send	VSS	SET PARAMETERS (CG1_reach:= 3; CG1_limit:= 2 000; CG1_decay:= 247/256; CG1_inc:= 25; L1:= 3)		Set the parameters to their default values.
Comments:						

Test Case Name:	UCTRL_param_M2inc					
Purpose:	**To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter M2inc.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	do		Antenna coupling such that receiver on GSC1 blocked by transmission on GSC2		Arrange the antenna coupling such that the receiver on GSC1 is blocked by any transmission on GSC2.
Test body	3	await	RF	SYNC_BURST_c (s= add_A)	Sc	Verify an autonomous sync burst is transmitted, indicating that the station is already on the network.
	4	send	RF (GSC1)	UCTRL_M2 (i:= 1; l:= 200; s:= add_G)	Cm	Send a broadcast UCTRL on GSC1 from a simulated ground station G to the station under test. The transmission sets M2limit to 200, and M2inc to 1.
	5	send	VSS (GSC2)	PERIODIC BROADCAST request to transmit SYNC_BURST_b (V11:= 60)	Sb	Establish sync burst transmissions on GSC2 at a rate of 1 per second.
	6	send	VSS (GSC1)	PERIODIC BROADCAST request to transmit SYNC_BURST_b (V11:= 60) with bursts midway between those on GSC2	Sb	Establish sync burst transmissions on GSC1 at a rate of 1 per second and midway between the bursts on GSC2.
	7	send	RF (GSC1)	UCTRL_M2 (i:= 120; l:= 200; s:= add_G)	Cm	Send a UCTRL on GSC1 from a simulated ground station G to the station under test. The transmission sets M2limit to 200, and M2inc to 120.
	8	record	RF (GSC1)	ctrl_time:= time at beginning of slot occupied by UCTRL_M2	Cm	
	9	verify	RF (GSC2)	Four SYNC_BURST_b (s= add_A) transmitted before time:= ctrl_time + 4	Sb	Wait for four sync bursts to be transmitted on GSC2 following the UCTRL. Four bursts transmitted on GSC2 should not cause network entry to be triggered.
	10	verify	RF (GSC1)	SYNC_BURST_b (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= ctrl_time + 5	Sb	Verify that a sync burst is transmitted on GSC1 following the four transmitted on GSC2 since the UCTRL, and that neither a BND nor a plea were transmitted. This verifies that network entry was not triggered on GSC1 after the first four sync bursts on GSC2. Network entry should be triggered in this second by the fifth burst being transmitted on GSC2.
	11	verify	RF (GSC1)	Before time:= ctrl_time + 66.68 EITHER no transmissions OR at least one plea transmitted in delayed burst OR at least one BND transmitted in delayed burst		Verify that network entry has been triggered on GSC1 by observing that for a period of M1 + 126 slots either: no transmissions are made by the station under test; or a BND has been transmitted; or a plea has been transmitted.
	12	wait		1 minute		
	13	send	RF (GSC1)	UCTRL_M2 (i:= 90; l:= 200; s:= add_G)	Cm	Send a broadcast UCTRL on GSC1 from a simulated ground station G to the station under test. The transmission sets M2limit to 200, and M2inc to 90.
	14	record	RF (GSC1)	ctrl_time:= time at beginning of slot occupied by UCTRL_M2	Cm	
	15	verify	RF (GSC2)	Twelve SYNC_BURST_b (s= add_A) transmitted before time:= ctrl_time + 12	Sb	Wait for 12 sync bursts to be transmitted on GSC2 following the UCTRL. 12 bursts transmitted on GSC2 should not cause network entry to be triggered.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	16	verify	RF (GSC1)	SYNC_BURST_b (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= ctrl_time + 13	Sb	Verify that a sync burst is transmitted on GSC1 following the 12 transmitted on GSC2 since the UCTRL, and that neither a BND nor a plea were transmitted. This verifies that network entry was not triggered on GSC1 after the first 12 sync bursts on GSC2. Network entry should be triggered in this second by the 13th burst being transmitted on GSC2.
	17	verify	RF (GSC1)	Before time:= ctrl_time + 74.68 EITHER no transmissions OR at least one plea transmitted in delayed burst OR at least one BND transmitted in delayed burst		Verify that network entry has been triggered on GSC1 by observing that for a period of M1 + 126 slots either: no transmissions are made by the station under test; or a BND has been transmitted; or a plea has been transmitted.
postamble	18	send	RF (GSC1)	UCTRL_M2 (i:= 2; l:= 160; s:= add_G)	Cm	Send a broadcast UCTRL from ground station G to the station under test, setting the parameters to their default values.
	19	send	VSS	CANCEL PERIODIC RESERVATION requests on GSC1 **AND** GSC2		Cancel established periodic streams.
	20	send	VSS	SET PARAMETERS (V11:= 1) on GSC1 **AND** GSC2		Reset to default value.
Comments:						

Test Case Name:	UCTRL_param_M2limit					
Purpose:	**To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter M2limit.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	do		Antenna coupling such that receiver on GSC1 blocked by transmission on GSC2		Arrange the antenna coupling such that the receiver on GSC1 is blocked by any transmission on GSC2.
Test body	3	await	RF	SYNC_BURST_c (s= add_A)	Sc	Verify an autonomous sync burst is transmitted, indicating that the station is already on the network.
	4	send	RF (GSC1)	UCTRL_M2 (i:= 1; l:= 280; s:= add_G)	Cm	Send a broadcast UCTRL on GSC1 from a simulated ground station G to the station under test. The transmission sets M2inc to 1, and M2limit to 360.
	5	send	VSS (GSC2)	PERIODIC BROADCAST request to transmit SYNC_BURST_b (V11:= 60) in the first slot of the UTC second	Sb	Establish sync burst transmissions on GSC2 at a rate of 1 per second in the first slot of the UTC second.
	6	send	VSS (GSC1)	PERIODIC BROADCAST request to transmit SYNC_BURST_b (V11:= 60) in the second slot of the UTC second	Sb	Establish sync burst transmissions on GSC1 at a rate of 1 per second in the second slot of the UTC second.
	7	send	RF (GSC1)	UCTRL_M2 (i:= 90; l:= 250; s:= add_G)	Cm	Send a broadcast UCTRL on GSC1 from a simulated ground station G to the station under test. The transmission sets M2inc to 90, and M2limit to 250.
	8	record	RF (GSC1)	ctrl_time:= time at beginning of slot occupied by UCTRL_M2	Cm	
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	9	verify	RF (GSC2)	Ten SYNC_BURST_b (s= add_A) transmitted before time:= ctrl_time + 9 + 1/M1	Sb	Wait for 10 sync bursts to be transmitted on GSC2 following the UCTRL. 10 bursts transmitted on GSC2 should not cause network entry to be triggered.
	10	verify	RF (GSC1)	BND not transmitted AND Plea not transmitted before time:= ctrl_time + 9 + 1/M1 AND SYNC_BURST_b (s= add_A) transmitted in slot beginning at time:= ctrl_time + 9 + 1/M1	Sb	Verify that a sync burst is transmitted on GSC1 following the 10 transmitted on GSC2 since the UCTRL, and that neither a BND nor a plea were transmitted. This verifies that network entry was not triggered on GSC1 after the first 10 sync bursts on GSC2.
	11	verify	RF (GSC2)	SYNC_BURST_b (s= add_A) transmitted in slot beginning at time:= ctrl_time + 10	Sb	Verify that the 11th sync burst is transmitted on GSC2. This causes network entry
	12	verify	RF (GSC1)	Before time:= ctrl_time + 71.68 EITHER no transmissions OR at least one plea transmitted in delayed burst OR at least one BND transmitted in delayed burst		Verify that network entry has been triggered on GSC1 by observing that for a period of M1 + 126 slots either: no transmissions are made by the station under test; or a BND has been transmitted; or a plea has been transmitted.
	13	wait		1 minute		

Test Case Name:	**UCTRL_param_M2limit**					
Purpose:	**To demonstrate that a station will act upon receipt of a UCTRL commanding ground modification of parameter M2limit.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	14	send	RF (GSC1)	UCTRL_M2 (i:= 490; l:= 90; s:= add_G)	Cm	Send a broadcast UCTRL on GSC1 from a simulated ground station G to the station under test. The transmission sets M2limit to 490, and M2inc to 90.
	15	record	RF (GSC1)	ctrl_time:= time at beginning of slot occupied by UCTRL_M2	Cm	
	16	verify	RF (GSC2)	Twenty-five SYNC_BURST_b (s= add_A) transmitted before time:= ctrl_time + 24 + 1/M1	Sb	Wait for 25 sync bursts to be transmitted on GSC2 following the UCTRL. 25 bursts transmitted on GSC2 should not cause network entry to be triggered.
	17	verify	RF (GSC1)	BND not transmitted AND Plea not transmitted before time:= ctrl_time + 24 + 1/M1 AND SYNC_BURST_b (s= add_A) transmitted in slot beginning at time:= ctrl_time + 24 + 1/M1	Sb	Verify that a sync burst is transmitted on GSC1 following the 25 transmitted on GSC2 since the UCTRL, and that neither a BND nor a plea were transmitted. This verifies that network entry was not triggered on GSC1 after the first 25 sync bursts on GSC2.
	18	verify	RF (GSC2)	SYNC_BURST_b (s= add_A) transmitted in slot beginning at time:= ctrl_time + 25	Sb	Verify that the 25th sync burst is transmitted on GSC2. This causes network entry
	19	verify	RF (GSC1)	Before time:= ctrl_time + 86.68 EITHER no transmissions OR at least one plea transmitted in delayed burst OR at least one BND transmitted in delayed burst		Verify that network entry has been triggered on GSC1 by observing that for a period of M1 + 126 slots either: no transmissions are made by the station under test; or a BND has been transmitted; or a plea has been transmitted.
postamble	20	send	RF (GSC1)	UCTRL_M2 (i:= 2; l:= 160; s:= add_G)	Cm	Send a broadcast UCTRL from ground station G to the station under test, setting the parameters to their default values.
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	21	send	VSS	CANCEL PERIODIC RESERVATION requests on GSC1 **AND** GSC2		Cancel established periodic streams.
	22	send	VSS	SET PARAMETERS (V11:= 1) on GSC1 **AND** GSC2		Reset to default value.
Comments:						

Test Case Name:	**Sync_Report_Receive**					
Purpose:	**To demonstrate that a station receiving a sequence of sync burst reports from a peer station will generate an appropriate output.**					
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.
test body	3	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(S 25 NM); lon:= CPR_LON(E 35 NM))	Sa	Send a sync burst from a simulated station B containing position information in the fixed data field.
	4	record	RF	sync_time:= time at start of slot containing sync burst		Define a reference time to measure relative times from during the test.
	5	rep 5		n:= 1; lat_data(n):= {CPR_LAT(S 30 NM), CPR_LAT(S 35 NM), CPR_LAT(S 40 NM), CPR_LAT(S 45 NM), CPR_LAT(S 50 NM)}; lon_data(n):= {CPR_LON(E 40 NM), CPR_LON(E 45 NM), CPR_LON(E 50 NM), CPR_LON(E 55 NM), CPR_LON(E 60 NM)}		Set up an array containing the sequence of positional data to be used in the test.
	6	await		time = sync_time + n × 30		
	7	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= lat_data(n); lon:= lon_data(n))	Sa	Send a sync burst containing the next position report in the sequence every 30 s for 5 minutes.
	8	record	AppOut	LAT DATA OUT, LON DATA OUT		Wait for the next received packet of data to be processed by the station and sent to the position output.
	9	endrep		n:= n + 1		Repeat for each report.
	10	verify	AppOut	LAT DATA OUT = {S 30 NM, S 35 NM, S 40 NM, S 45 NM, S 50 NM} AND LON DATA OUT = {E 40 NM, E 45 NM, E 50 NM, E 55 NM, E 60 NM}		Verify that the station under test generates the appropriate output.
postamble	11	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**Sync_Report_Simultaneous**					
Purpose:	**To demonstrate that a station is capable of receiving sync burst reports simultaneously on both GSCs.**					
Special test instructions:	This test case tests the requirement of clause 6.3.10 in EN 302 842-1 [4]. The test description is located in the present document and not in EN 302 842-1 [**4**] as the present document includes all the necessary additional requirements and information for the completion of protocol tests, of which this is one.					
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.
test body	3	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(0)) on GSC 1 AND SYNC_BURST_a (pt:= 3; po:= 0; s:= add_C; lat:= CPR_LAT(N 10 NM); lon:= CPR_LON(E 10 NM)) on GSC 2	Sa	Send a sync burst from a simulated station B on GSC 1 and from simulated station C in the same slot on GSC 2, both containing position information in the fixed data fields.
	4	await	AppOut	LAT DATA OUT B, LON DATA OUT B AND LAT DATA OUT C, LON DATA OUT C		Wait for the received reports from stations B and C to be processed by the station and sent to the position output.
	5	verify	AppOut	LAT DATA OUT B = 0 AND LON DATA OUT B = 0 AND LAT DATA OUT C = N 10 NM AND LON DATA OUT C = E 10 NM		Verify that the station under test processes the data and generates the appropriate output.
postamble	6	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	**CPR_Encode**					
Purpose:	**To demonstrate that a series of latitude and longitude positions may be correctly encoded in the sync burst 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.
	4	send	VSS	REQUEST TO TRANSMIT SYNC_BURST (V11:= 60)		Set the station under test to transmit sync bursts at the rate of 1 per second.
test body	5	rep 2166		n:= 1; initialize p		
	6	send	Position	Input to station under test: LAT(n):= 12.8557 + n × 0.0163 LON(n):= -0.8150 + n × 0.0163	CE(r, c)	Send test values of latitude and longitude from CPR_ENC_TABLE to the station under test.
	7	rep 135		k:= 1		
	8	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		
	9	endrep		k:= k + 1		
	10					
	11	await	RF	SYNC_BURST_I (s= add_A)	SI	
	12	do		IF cprf in fixed part of SYNC_BURST_I (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	SI	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.
	13	 verify verify verify	 RF RF RF	In fixed part of SYNC_BURST_I (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)	SI, 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.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	14	endrep		n:= n + 1		
postamble	15	send	VSS	SET PARAMETERS (p:= 64/256; V11:= 1)		Restore to default value.
	16	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						
NOTE: The test values provided in the CPR_ENC_TABLE can only be used if the CPR type happens to correspond to the type for which the test values were calculated. If this is not the case when the n test loop starts for the first time, the n test loop must be restarted until this happens.						

Test Case Name:	**CPR_Decode**					
Purpose:	**To demonstrate that a series of latitude and longitude positions may be correctly decoded from the sync burst 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 (L1:= 10)		Set the maximum number of missed reservations to 10.
test body	4	rep 135		n:= 1		
	5	send	Position	Input to station under test: LAT(n):= CPR_DEC_TABLE (n, lat sut) LON(n):= CPR_DEC_TABLE (n, lon sut)	CD(r, c)	Input the position of the station under test.
	6	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)	SI, CE(r, c)	Send a sync burst from a simulated station B. The encoded values for lat and lon in the fixed part of the burst are taken from row n of CPR_ENC_TABLE.
	7	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 application output.
	8	verify	AppOut	LAT DATA OUT = CPR_DEC_TABLE (n, decoded lat) AND LON DATA OUT = CPR_DEC_TABLE (n, decoded lon)	CD(r, c)	Verify that the station under test processes the data and generates the appropriate output for display to the aircrew.
	9	endrep		n:= n + 1		
postamble	10	send	VSS	SET PARAMETERS (L1:= 3)		Restore to default value.
	11	send	VSS	REINSTATE AUTONOMOUS SYNC BURSTS		Reinstate the autonomous sync bursts.
Comments:						

Test Case Name:	Power_Interrupt_A					
Purpose:	**To demonstrate that a station continues to operate the receiver through a short power interrupt not exceeding Y ms.**					
Special test instructions:	This test case tests the requirement of clause 6.3.5 in EN 302 842-1 [**4**]. The test description is located in the present document and not in EN 302 842-1 [4] as the present document includes all the necessary additional requirements and information for the completion of protocol tests, of which this is one. The value of Y in this test is determined by the value set in ED-14D/RTCA DO-160D [**11**] for the appropriate equipment category (see clause 6.3.5 of EN 302 842-1 [**4**]).					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS (p:= 1)		Ensure 100 % chance of transmission on access.
test body	3	await	RF	SYNC_BURST_c (s= add_A)	Sc	Wait for an autonomous sync burst to be transmitted.
	4	do		Remove power from VDL4 transceiver		Remove power from receiver.
	5	send	RF	SYNC_BURST_b (pt:= 3; s:= add_B)	Sb	Send a sync burst with pt = 3 from a simulated station B, reserving a stream of slots.
	6	record		sync_time:= time at beginning of slot occupied by SYNC_BURST_b	Sb	
	7	do		Restore power to VDL4 transceiver within Y ms of power removal		Restore power to the receiver within Y ms.
	8	macro		M_RAND_ACC_SU (sf:= 3)		Queue random access transmissions over 3 superframes.
	9	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions
	10	record		rand_time:= start of random access transmissions num_slots:= (rand_time - sync_time) × M1/60		
	11	rep 3×M1		n:= 1		Verify over 3 superframes.
	12	verify	RF	IF n + num_slots = {M1, 2×M1, 3×M1} THEN no RAND_ACC_DATA_a (s= add_A) in slot beginning at time = sync_time + n × 60	Ra	Verify that no random access transmissions are made by the station under test in a slot reserved by station B over 3 superframes.
	13	endrep		n:= n + 1		
postamble	14					Bring test equipment into idle state.
Comments:						

Test Case Name:	**Power_Interrupt_B**					
Purpose:	**To demonstrate that a station will observe the m2 filter after a long power interrupt (exceeding Y ms).**					
Special test instructions:	The value of Y is the maximum period of a short power interrupt as determined by the value set in ED-14D/RTCA DO-160D [11] for the appropriate equipment category (see clause 6.3.5 of EN 302 842-1 [**4**]). It is assumed that Y is less than the period of power interrupt used in this test.					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	send	VSS	SET PARAMETERS (p:= 1)		Ensure 100 % chance of transmission on access.
	3	do		NETWORK ENTRY BY PERIODIC AND INCREMENTAL		Ensure transceiver is set to perform network entry by a combination of periodic and incremental broadcasts as opposed to other means.
test body	4	send	RF	UCTRL_M2 (i:= 1; l:= 400; s:= add_G)	Cm	Send a broadcast UCTRL from a simulated ground station G to the station under test. The transmission sets M2limit to 400, and M2inc to 1.
	5	send	RF	SYNC_BURST_b (pt:= 3; s:= add_B)	Sb	Send a sync burst with pt = 3 from a simulated station B reserving a stream of slots.
	6	record		sync_time:= time at beginning of slot occupied by SYNC_BURST_b	Sb	
	7	do		Remove power from VDL4 transceiver for 5 s.		Simulate a power interruption of 5 seconds (assumed to be longer than the maximum period of a short power interrupt Y).
	8	do		Restore power to VDL4 transceiver		
	9	macro		M_RAND_ACC_SU (sf:= 3)		Queue random access transmissions over 3 superframes.
	10	await	RF	RAND_ACC_DATA_a (s= add_A)	Ra	Wait for the start of the random access transmissions
	11	record		rand_time:= start of random access transmissions num_slots:= (rand_time - sync_time) × M1/60		
	12	rep 3×M1		n:= 1		Verify over 3 superframes.
	13	verify	RF	IF n + num_slots = {M1, 2×M1, 3×M1} THEN No RAND_ACC_DATA_a (s= add_A) in slot beginning at time = sync_time + n × 60	Ra	Verify that no random access transmissions are made by the station under test in a slot reserved by station B over 3 superframes.
	14	endrep		n:= n + 1		
	15	wait		1 minute		
	16	send	VSS	PERIODIC BROADCAST request to transmit SYNC_BURST_b (V11:= 60) in the first slot of the UTC second	Sb	Establish sync burst transmissions from the station under test at a rate of 1 per second in the first slot of the UTC second.
	17	do		Remove power from VDL4 transceiver for 5,4 seconds.		Remove electrical power from the transceiver for 5,4 seconds.
	18	do		Restore power to VDL4 transceiver		Restore electrical power.
	19	record		on_time:= time at restoration of power		Record the time that power was restored.
	20	verify	RF	No transmissions from the station under test before time:= on_time + 60		Verify that there are no transmissions by the station under test for a period of one minute, in which time the station shall be listening to the channel to build up a complete slot map.

Context	Step	Action	PCO	Action Qualifier	Ref	Comment
	21	await	RF (GSC1)	SYNC_BURST_c (s= add_A) transmitted at or after time:= on_time + 60	Sc	Verify an autonomous sync burst is then transmitted.
	22	verify	RF (GSC1)	SYNC_BURST_c (s= add_A) contains pt = 3 AND io ≠ 0 (or po ≠ 0)	Sc	Verify that the first sync burst transmitted contains pt and io (or po) values compatible with a combined periodic and incremental broadcast reservation.
postamble	23					Bring test equipment into idle state.
Comments:						

Test Case Name:	DLS_NotSupported					
Purpose:	**To demonstrate that a station in receipt of a CTRL_RTS transmits a general failure with an error type of 80 hex when it does not support the DLS.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
	2	do		CONFIGURE TO NOT SUPPORT DLS		Configure the equipment under test so that it does not support the DLS.
test body	3	send	RF	CTRL_RTS_a (s:= add_B; d:= add_A; IB:= 1; T:= 0; pr:= 3; lg:= 7)	CRa	Send an RTS using the long transmission procedures from a simulated station B, with IB = 1 and T = 0, indicating this is the first transmission from station B to station A. The RTS contains a unicast reservation for a response.
	4	await		BURST transmitted by station A		Wait for a burst transmitted by station A.
	5	verify	RF	GEN_RESP_a (s = add_A; d = add_B; r-mi = 0001101; ok = 0; err = 80 hex) in the slot reserved by the RTS	GRa	Verify that a general failure, with error type set to 80 hex, is transmitted by station A in the slot reserved by the RTS.
postamble	6					Bring test equipment into idle state.
Comments: This test is optional. A station that implements the DLS should not perform this test.						

Test Case Name:	**DLS_UDATA_Receive**					
Purpose:	**To demonstrate that a UDATA DLPDU received from another station will be forwarded to the DLS user.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	send	RF	UCTRL_a (s:= add_B)	UCa	Send a UDATA DLPDU to the station under test.
	3	verify	DLS	data in UCTRL_a (s:= add_B) passed to DLS user	UCa	Verify that the data in the UDATA DLPDU is passed to the DLS user.
	4	verify	RF	No ACK transmitted by station A		Verify that the station under test does not generate an ACK in response to the UDATA DLPDU.
	5	send	RF	UINFO_a (s:= add_B)	UIa	Send a UDATA DLPDU to the station under test.
	6	verify	DLS	data in UINFO_a (s:= add_B) passed to DLS user	UIa	Verify that the data in the UDATA DLPDU is passed to the DLS user.
	7	verify	RF	No ACK transmitted by station A		Verify that the station under test does not generate an ACK in response to the UDATA DLPDU.
postamble	8					Bring test equipment into idle state.
Comments:						

Test Case Name:	**DLS_UDATA_Invalid**					
Purpose:	**To demonstrate that a station receiving an invalid UDATA DLPDU will detect and discard it.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	send	RF	UINFO_b (s:= add_B)	UIb	Send a UDATA DLPDU by the short transmission procedure to the station under test. The UDATA burst has no information field and bit 8 of octet 5 is not present. The burst contains a response reservation.
	3	verify	DLS	UINFO_b has been discarded by the station under test	UIb	Verify at the DLS that the incorrectly coded message has been discarded by the station under test. It is assumed the MAC layer would not detect and discard an error of this kind.
postamble	4					Bring test equipment into idle state.
Comments:						

Test Case Name:	Param_L1					
Purpose:	**To demonstrate that a station will act correctly upon modification of private parameter L1.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
preamble	1	do		M_POWER_UP		Prepare the transceiver for testing.
test body	2	await	RF	SYNC_BURST_c (s= add_A)	Sc	Verify an autonomous sync burst is transmitted, indicating that the station is already on the network.
	3	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 5; CG1_decay:= 1; CG1_inc:= 1; L1:= 2)		Set CG1_reach to 1, CG1_limit to 5, CG1_decay to its maximum value (minimum rate of decay), CG1_inc to 1 and L1 to 1 while leaving the other parameters at their default values.
	4	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM)) (position of station B is < CG1_range away from station under test)	Sa	Send a sync burst from a simulated station B < CG1_range away from the station under test. The purpose of sending this sync burst is so that station B will not have been unreachable for > CG1_reach minutes when the next sync burst is received.
	5	record	RF	sync_time1:= time at beginning of slot occupied by SYNC_BURST_a	Sa	
	6	verify	RF	SYNC_BURST_c (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= sync_time1 + 60	Sc	Verify that network entry has not been triggered by observing that for a period of one minute: an autonomous transmission is made by the station under test; and a BND has not been transmitted; and a plea has not been transmitted.
	7	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 5; CG1_decay:= 1; CG1_inc:= 1; L1:= 1)		Set CG1_reach to 1, CG1_limit to 5, CG1_decay to its maximum value (minimum rate of decay), CG1_inc to 1 and L1 to 1 while leaving the other parameters at their default values.
	8	await		time = sync_time1 + 170		Wait 170 s since the last sync burst from station B.
	9	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM)) (position of station B is < CG1_range away from station under test) in slot beginning at time = sync_time1 + 170	Sa	Send a sync burst from station B < CG1_range away from the station under test.
	10	record	RF	sync_time2:= time at beginning of slot occupied by SYNC_BURST_a	Sa	
	11	verify	RF	SYNC_BURST_c (s= add_A) transmitted AND BND not transmitted AND Plea not transmitted before time:= sync_time2 + 60	Sc	Verify that network entry has not been triggered by observing that for a period of one minute: an autonomous transmission is made by the station under test; and a BND has not been transmitted; and a plea has not been transmitted.

Test Case Name:	**Param_L1**					
Purpose:	**To demonstrate that a station will act correctly upon modification of private parameter L1.**					
Context	**Step**	**Action**	**PCO**	**Action Qualifier**	**Ref**	**Comment**
	12	send	VSS	SET PARAMETERS (CG1_reach:= 1; CG1_limit:= 5; CG1_decay:= 1; CG1_inc:= 1; L1:= 2)		Set CG1_reach to 1, CG1_limit to 5, CG1_decay to its maximum value (minimum rate of decay), CG1_inc to 10 and L1 to 2 while leaving the other parameters at their default values.
	13	await		time = sync_time2 + 170		Wait 170 s since last sync burst from station B.
	14	send	RF	SYNC_BURST_a (pt:= 3; po:= 0; s:= add_B; lat:= CPR_LAT(0); lon:= CPR_LON(E 70 NM)) (position of station B is < CG1_range away from station under test) in slot beginning at time = sync_time2 + 170	Sa	Send a sync burst from station B < CG1_range away from the station under test.
	15	verify	RF	Before time:= sync_time2 + 170 + 61.68 EITHER No transmissions OR Plea transmitted in delayed burst OR BND transmitted in delayed burst		Verify that network entry has been triggered by observing that for a period of M1 + 126 slots either: no transmissions are made by the station under test; or a BND has been transmitted; or a plea has been transmitted.
postamble	16	send	VSS	SET PARAMETERS (CG1_reach:= 3; CG1_limit:= 2 000; CG1_decay:= 247/256; CG1_inc:= 25; L1:= 3)		Set parameters to their default values.
Comments:						

Annex A (informative): Cross reference matrix

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

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

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

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

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

NOTE 3: The requirement listed in column 1 is applicable only to VDL Mode 4 ground equipment. No 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: The ICAO requirement listed in Column 1 is reflected in a more specific EN requirement specified and tested elsewhere in the present document.

NOTE 6: The ICAO requirement listed in Column 1 is outside the scope of a unit supporting only the minimum core functionality as defined here.

NOTE 7: Only part of the ICAO requirement listed in Column 1 has been listed here, as the other part is outside the scope of a unit supporting only the minimum core functionality as defined here.

NOTE 8: 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" shall be performed as specified with satisfactory outcomes as a necessary condition for a presumption of conformity. Requirements associated with tests classified "O" or "X" must be complied with as a necessary condition for presumption of conformity, although conformance with the requirement may be claimed by an equivalent test or by manufacturer's assertion supported by appropriate entries in the technical construction file.

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

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.1	1.2		MAC sublayer.	See note 1	X
5.1.1			Services	See note 1	X
5.1.1.1	1.2	a		See note 2	O
5.1.2	1.2.1		MAC sublayer services.	See note 1	X
5.1.2.1	1.2.1.2	a		See note 2	O
5.1.2.2	1.2.1.2	b		See note 2	O
5.1.2.3	1.2.1.2	c		See note 2	O
5.1.3	1.2.2		MAC sublayer parameters.	See note 1	X
5.1.3.1			General	See note 1	X
5.1.3.1.1	1.2.2	a		See note 1a	X
5.1.3.2	1.2.2.1		Parameter M1 (number of slots per superframe).	See note 1	X
5.1.3.2.1	1.2.2.1	a		See note 1a	X
5.1.3.2.2	1.2.2.1	b		See note 1a	X
5.1.3.3	1.2.2.2		Parameter M2inc and M2limit (MAC layer control parameters for network entry)	See note 1	X
5.1.3.3.1	1.2.2.2.1	a		UCTRL_param_M2inc UCTRL_param_M2limit Power_Interrupt_B	E
5.1.3.3.2	1.2.2.2.1	b		UCTRL_param_M2inc UCTRL_param_M2limit Power_Interrupt_B	E
5.1.3.3.3	1.2.2.2.2	a		UCTRL_param_M2limit	E
5.1.3.4			Inspection of m2 filter following transceiver failure	See note 1	X
5.1.3.4.1				Power_Interrupt_B	E
5.1.4	1.2.3		Time synchronization.	See note 1	X
5.1.4.1	1.2.3.1		Primary.	See note 1	X
5.1.4.1.1	1.2.3.1	a		Timing_Primary	E
5.1.4.2	1.2.3.2		Secondary.	See note 1	X
5.1.4.2.1	1.2.3.2	a		Timing_Secondary	E
5.1.4.2.2	1.2.3.2	b		Timing_Secondary	E
5.1.4.2.3	1.2.3.2	c		Timing_Secondary_Recover	E
5.1.4.2.4				See note 2	O
5.1.4.2.5				See note 2	O
5.1.4.3	1.2.3.3		Alignment to UTC second.	See note 1	X
5.1.4.3.1	1.2.3.3	a		See note 1a	X
5.1.4.4	1.2.3.5		Data quality level.	See note 1	X
5.1.4.4.1	1.2.3.5.1	a		See note 1a	X
5.1.4.4.1a	1.2.3.5.1	a		See note 2	O
5.1.4.4.2	1.2.3.5.1	a		See note 1a	X
5.1.4.4.3				See note 2	O
5.1.5	1.2.4		Slot idle/busy notification.	See note 1	X
5.1.5.1	1.2.4.1		Slot idle detection.	See note 1	X
5.1.5.1.1	1.2.4.1	a		See note 1a	X
5.1.5.2	1.2.4.2		Slot busy detection.	See note 1	X
5.1.5.2.1	1.2.4.2	a		See note 1a	X
5.1.5.3	1.2.4.3		Slot occupied detection.	See note 1	X
5.1.5.3.1	1.2.4.3	a		See note 1a	X
5.1.5.4	1.2.4.4		Signal level indication.	See note 1	X
5.1.5.4.1	1.2.4.4	a		See note 2	O
5.1.6	1.2.5		Transmission processing.	See note 1	X
5.1.6.1	1.2.5.1	a		See note 2	O
5.1.6.2	1.2.5.2	a		Slot_Boundary	E
5.1.6.3	1.2.5.2	b		Slot_Delayed	E
5.1.7	1.2.6		Received transmission processing.	See note 1	X
5.1.7.1	1.2.6	a		CRC_Rej	E
5.1.7.2	1.2.6	b		Periodic_NonDitherRes Sync_Report_Receive	E
5.2	1.3		VSS sublayer.	See note 1	X
5.2.1	1.3.1		Services.	See note 1	X
5.2.1.a	1.3.1.1		Multiple access	See note 1	X

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.2.1.a.1		a		See note 1a	X
5.2.1.1	1.3.1.2		Error detection.	See note 1	X
5.2.1.1.1	1.3.1.2	a		CRC_Norm	E
5.2.1.2	1.3.1.3		Channel congestion.	See note 1	X
5.2.1.2.1	1.3.1.3	a		See note 2	O
5.2.2	1.3.2		Burst format.	See note 1	X
5.2.2.1			VSS burst structure	See note 1	X
5.2.2.1.1	1.3.2	a		Sync_Format	E
5.2.2.1.2				See note 1a	X
5.2.2.2	1.3.2.1		Version number.	See note 1	X
5.2.2.2.1	1.3.2.1	a		See note 1a	X
5.2.2.2.2	1.3.2.1	b		Sync_Format	E
5.2.2.2.3	1.3.2.1	c		Version_NonZero	E
5.2.2.3	1.3.2.2		Source address.	See note 1	X
5.2.2.3.1	1.3.2.2	a		See note 1a	X
5.2.2.3.2	1.3.2.2	b		See note 1a	X
5.2.2.4	1.3.2.3		Message ID.	See note 1	X
5.2.2.4.1	1.3.2.3.1	a		See note 1a	X
5.2.2.4.2	1.3.2.3.1	b		See note 1a	X
5.2.2.4.3	1.3.2.3.2	a		See note 2	O
5.2.2.5	1.3.2.4		Information field.	See note 1	X
5.2.2.5.1	1.3.2.4	a		See note 2	O
5.2.2.6	1.3.2.5		Reservation fields.	See note 1	X
5.2.2.6.1	1.3.2.5	a		See note 1a	X
5.2.2.6.2	1.3.2.5	b		See note 1a	X
5.2.2.6.3	1.3.2.5	c		See note 1a	X
5.2.2.7	1.3.2.6		Autonomous/directed flag.	See note 1	X
5.2.2.7.1	1.3.2.6.1	a		See note 2	O
5.2.3	1.3.3		VSS sublayer parameters.	See note 1	X
5.2.3.1			General	See note 1	X
5.2.3.1.1	1.3.3	a		See note 2	O
5.2.3.2	1.3.3.1		Parameter VS1 (number of ground quarantined slots).	See note 1	X
5.2.3.2.1	1.3.3.1	a		See note 1a	X
5.2.3.2.2				See note 1	X
5.2.3.2.3	1.3.6.4.1 1.3.6.4.2 1.3.6.4.3	a,b a,b a		Periodic_Quarantine_A Periodic_Quarantine_B Autotune_DirectedQuarantine UCTRL_param_VS1 UCTRL_param_VS4 Periodic_CancelQuarantine Block_Superframe_Quarantine Block_Superframe_Reserve_A Block_Superframe_Reserve_B Block_Superframe_Reserve_C Block_Secondframe_Reserve	E
5.2.3.3	1.3.3.2		Parameter VS2 (minimum CCI performance).	See note 1	X
5.2.3.3.1	1.3.3.2	a		See note 1a	X
5.2.3.3.2	1.3.3.2	b		See note 1a	X
5.2.3.4	1.3.3.3		Parameter VS4 (quarantine slot re-use range).	See note 1	X
5.2.3.4.1	1.3.3.3	a		See note 1a	X
5.2.3.5	1.3.3.4		Parameter VS5 (maximum burst length)	See note 1a	X
5.2.3.5.1	1.3.3.4	a		See note 1a	X
5.2.4	1.3.4		VSS quality of service parameters.	See note 1	X
5.2.4.1			General	See note 1	X
5.2.4.1.1	1.3.4	a		See note 2	O
5.2.4.2	1.3.4.1		Parameter Q1 (priority).	See note 1	X
5.2.4.2.1	1.3.4.1	a		See note 1a	X
5.2.4.3	1.3.4.2		Parameters Q2a to Q2d (slot selection range constraint for level n).	See note 1	X
5.2.4.3.1	1.3.4.2	a		See note 1a	X
5.2.4.3.2	1.3.6.2.2.1	h		See note 1a	X
5.2.4.4	1.3.4.3		Parameter Q3 (replace queued data).	See note 1	X
5.2.4.4.1	1.3.4.3	a		See note 1a	X

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.2.4.4.2	1.3.4.3	b		Queue_Replace	E
5.2.4.4.3	1.3.4.3	c		Queue_Norm	E
5.2.4.5	1.3.4.4		Parameter Q4 (number of available slots).	See note 1	X
5.2.4.5.1	1.3.4.4	a		See note 1a	X
5.2.5	1.3.5		Received transmission processing.	See note 1	X
5.2.5.1	1.3.5.5	a		See note 2	O
5.2.5.2	1.3.5.5	b		See note 2	O
5.2.5.3	1.3.5.1	a		Periodic_NonDitherRes Periodic_DitherRes Periodic_Replacement Periodic_Cancel Incremental_Reservation_A Unicast_Reservation_A Info_Reservation Autotune_Autonomous_A Autotune_Autonomous_B Autotune_Autonomous_C Autotune_Autonomous_D Autotune_Autonomous_E Autotune_Directed_A Autotune_Directed_B Autotune_Reservation Autotune_CancelResp Autotune_CancelAbsent Autotune_Override Autotune_DirectedQuarantine	E
5.2.5.4	1.3.5.1	b		Reservation_Unrecognized	E
5.2.5.5	1.3.5.2	a		Reservation_Invalid	E
5.2.5.6	1.3.5.3	a		See note 2	O
5.2.5.7	1.3.5.4	a		See note 1a	X
5.2.5.8	1.3.5.4	b		See note 1a	X
5.2.5.9	1.3.5.5	c		MessageID_Invalid_A MessageID_Invalid_B	E
5.2.6	1.3.6		Reserved access protocol specification.	See note 1	X
5.2.6.1	1.3.6.1		Reservation table.	See note 1	X
5.2.6.1.1	1.3.6.1.1	a		Periodic_NonDither_Res Periodic_DitherRes Autotune_Reservation Incremental_Reservation_A Unicast_Reservation_A Info_Reservation	E
5.2.6.1.2	1.3.6.1.1	b		Periodic_NonDither_Res Periodic_DitherRes Autotune_Reservation Incremental_Reservation_A Unicast_Reservation_A Info_Reservation	E
5.2.6.1.3	1.3.6.1.1	c		Periodic_NonDither_Res Periodic_DitherRes Autotune_Reservation	E
5.2.6.1.4	1.3.6.1.2	a		Periodic_Potential	E
5.2.6.1.5	1.3.6.1.2	b		See note 2	O
5.2.6.1.6	1.3.6.1.2	c		See note 2	O
5.2.6.1.7	1.3.6.1.3	a		Reservation_Recognition	E
5.2.6.1.8	1.3.6.1.4	a		NetEntry_OneMinute	E
5.2.6.2	1.3.6.2		Selecting slots for transmission or reservation.	See note 1	X

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.2.6.2.1	1.3.6.2	a		SlotSel_Level0_A SlotSel_Level0_B SlotSel_Level0_C SlotSel_Level0_D SlotSel_Level0_E SlotSel_Level0_F SlotSel_Level1_A SlotSel_Level1_B SlotSel_Level1_C SlotSel_Level1_D SlotSel_Level1_E SlotSel_Level1_F SlotSel_Level2_A SlotSel_Level2_B SlotSel_Level2_C SlotSel_Level2_D SlotSel_Level2_E SlotSel_Level3_A SlotSel_Level3_B SlotSel_Level3_C SlotSel_Level3_D SlotSel_Level4_A SlotSel_Level4_B SlotSel_Level4_C SlotSel_Unsuccessful	E
5.2.6.2.2	1.3.6.2.1	b		See note 2	O
5.2.6.2.3	1.3.6.2.1	c		SlotSel_Level0_A SlotSel_Level0_B SlotSel_Level0_C SlotSel_Level0_D SlotSel_Level0_E SlotSel_Level0_F SlotSel_Level1_A SlotSel_Level1_B SlotSel_Level1_C SlotSel_Level1_D SlotSel_Level1_E SlotSel_Level1_F SlotSel_Level2_A SlotSel_Level2_B SlotSel_Level2_C SlotSel_Level2_D SlotSel_Level2_E SlotSel_Level3_A SlotSel_Level3_B SlotSel_Level3_C SlotSel_Level3_D SlotSel_Level4_A SlotSel_Level4_B SlotSel_Level4_C SlotSel_Unsuccessful	E
5.2.6.2.4	1.3.6.2.1	d		SlotSel_QoSGroup	E
5.2.6.2.5	1.3.6.2.1	e		SlotSel_Unsuccessful	E
5.2.6.2.6	1.3.6.2.1	a		See note 2	O

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.2.6.2.7	1.3.6.2.2.1	a		SlotSel_Level0_A SlotSel_Level0_B SlotSel_Level0_C SlotSel_Level0_D SlotSel_Level0_E SlotSel_Level0_F SlotSel_Level1_A SlotSel_Level1_B SlotSel_Level1_C SlotSel_Level1_D SlotSel_Level1_E SlotSel_Level1_F SlotSel_Level2_A SlotSel_Level2_B SlotSel_Level2_C SlotSel_Level2_D SlotSel_Level2_E SlotSel_Level3_A SlotSel_Level3_B SlotSel_Level3_C SlotSel_Level3_D SlotSel_Level4_A SlotSel_Level4_B SlotSel_Level4_C	E
5.2.6.2.8	1.3.6.2.2.1	b		"	E
5.2.6.2.8a				"	E
5.2.6.2.9	1.3.6.2.2.1	c		"	E
5.2.6.2.10	1.3.6.2.2.1	d		"	E
5.2.6.2.11	1.3.6.2.2.1	e		"	E
5.2.6.2.12	1.3.6.2.2.1	f		"	E
5.2.6.2.13	1.3.6.2.2.1	g		"	E
5.2.6.2.14	1.3.6.2.2.2.1	a		See note 4	O
5.2.6.2.14a	1.3.6.2.2..3	a		SlotSel_Exclusion	E
5.2.6.2.14b	1.3.6.2.3	a		Block_SecondFrame_Reserve	E
5.2.6.2.14c	1.3.6.2.3	b		See note 3	X
5.2.6.2.15	1.3.6.2.4	a		SlotSel_Unsuccessful	E
5.2.6.2.16	1.3.6.2.4	b		SlotSel_Level0_A SlotSel_Level0_B SlotSel_Level0_C SlotSel_Level0_D SlotSel_Level0_E SlotSel_Level0_F SlotSel_Level1_A SlotSel_Level1_B SlotSel_Level1_C SlotSel_Level1_D SlotSel_Level1_E SlotSel_Level1_F SlotSel_Level2_A SlotSel_Level2_B SlotSel_Level2_C SlotSel_Level2_D SlotSel_Level2_E SlotSel_Level3_A SlotSel_Level3_B SlotSel_Level3_C SlotSel_Level3_D SlotSel_Level4_A SlotSel_Level4_B SlotSel_Level4_C	E
5.2.6.2.17	1.3.6.2.5	a		SlotSel_Block_Level0_A SlotSel_Block_Level0_B SlotSel_Block_MixedLevel	E
5.2.6.2.18	1.3.6.2.5	b		SlotSel_Block_MixedLevel	E

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.2.6.2.19	1.3.6.2.5	c		SlotSel_Block_Level0_A SlotSel_Block_Level0_B SlotSel_Block_MixedLevel	E
5.2.6.2.20	1.3.6.2.6	a		SlotSel_Reselection	E
5.2.6.3	1.3.6.3		Reserved transmissions.	See note 1	X
5.2.6.3.1	1.3.6.3	a		See note 2	O
5.2.6.3.2	1.3.6.3.1	a		See note 2	O
5.2.6.3.3	1.3.6.3.2	a		See note 2	O
5.2.6.4	1.3.6.5		Reservation conflicts.	See note 1	X
5.2.6.4.1	1.3.6.5	a		Conflict_Periodic_A Conflict_Periodic_B Conflict_Periodic_C Conflict_NoAction Conflict_Incremental Conflict_Priority Conflict_FirstRequest	E
5.2.6.4.2	1.3.6.5	b		Conflict_Priority Conflict_FirstRequest	E
5.2.6.4.3	1.3.6.5	c		See note 2	O
5.2.6.4.4	1.3.6.5	d		Conflict_Directed	E
5.2.6.4.5	1.3.6.5	e		See note 2	O
5.2.6.4.6	1.3.6.5	f		Conflict_NoAction	E
5.2.6.4.7	1.3.6.5	g		Conflict_Periodic_A Conflict_Periodic_B Conflict_Periodic_C Conflict_Incremental Conflict_BND	E
5.2.6.5	1.3.6.6		Transmission conflicts for mobile stations	See note 1	X
5.2.6.5.1	1.3.6.6	a		Conflict_Channel_Priority Conflict_Channel_Ground_A Conflict_Channel_Ground_B Conflict_Channel_FirstRequest	E
5.2.6.5.2	1.3.6.6	b		Conflict_Channel_Priority	E
5.2.6.5.3	1.3.6.6	c		Conflict_Channel_Ground_A	E
5.2.6.5.4	1.3.6.6	d		Conflict_Channel_Ground_B	E
5.2.6.5.5	1.3.6.6	e		Conflict_Channel_FirstRequest	E
5.2.7	1.3.7		Random access protocol specification.	See note 1	X
5.2.7.1			General	See note 1	X
5.2.7.1.1	1.3.7	a		Rand_Persistence	E
5.2.7.2	1.3.7.1		Random access parameters.	See note 1	X
5.2.7.2.1	1.3.7.1	a		See note 1a	X
5.2.7.2.2	1.3.7.1.1	a		See note 1a	X
5.2.7.2.3	1.3.7.1.1	b		Rand_Congestion	E
5.2.7.2.4	1.3.7.1.1	c		Rand_TM2Clear Rand_TM2Reset	E
5.2.7.2.5	1.3.7.1.1	d		Rand_Congestion	E
5.2.7.2.6	1.3.7.1.2	a		See note 1a	X
5.2.7.2.7	1.3.7.2.1.3	a		Rand_Persistence	E
5.2.7.2.8	1.3.7.1.3	a		Rand_MaxAttempts	E
5.2.7.2.9	1.3.7.1.3	b		Rand_MaxAttempts Rand_VS3Clear	E
5.2.7.2.10	1.3.7.1.3	c		Rand_MaxAttempts	E
5.2.7.2.11	1.3.7.1.3	d		Rand_MaxAttempts	E
5.2.7.3	1.3.7.2		Random access procedures.	See note 1	X
5.2.7.3.1	1.3.7.2.1.1	a		Rand_Persistence	E
5.2.7.3.2	1.3.7.2.1.1	b		Periodic_DitherRes Incremental_Reservation_A Unicast_Reservation_A Info_Reservation Autotune_Reservation Slot_Boundary	E
5.2.7.3.3	1.3.7.2.1.1	c		Rand_Availability	E
5.2.7.3.4	1.3.7.2.1.1	d		Rand_Busy	E
5.2.7.3.5	1.3.7.2.1.2	a		Rand_Congestion	E

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.2.7.3.5a	1.3.7.2.2.1	a		Rand_Persistence Slot_Delayed	E
5.2.7.3.5b	1.3.7.2.2.1	b		Slot_Delayed	E
5.2.7.3.5c	1.3.7.2.2.1	c		NetEntry_Delayed_Plea	E
5.2.7.3.5d	1.3.7.2.2.2	a		Rand_Congestion	E
5.2.7.3.5e	1.3.7.2.2.3	a		Rand_Persistence	E
5.2.7.3.6	1.3.7.2.3.1	a		See note 4	O
5.2.7.3.7	1.3.7.2.3.1	b		See note 4	O
5.2.7.3.10	1.3.7.2..4	a		See note 2	O
5.2.7.3.11	1.3.7.2..4	b		Rand_Priority	E
5.2.7.3.12	1.3.7.2..4	c		Queue_Replace Queue_Norm	E
5.2.9	1.3.9		Null reservation protocol specification.	See note 1	X
5.2.9.1	1.3.9.1		Null reservation burst format.	See note 1	X
5.2.9.1.1	1.3.9.1	a		Null_Reservation	E
5.2.9.1.2	1.3.9.1	b		See note 1a	X
5.2.10	1.3.10		Periodic broadcast protocol specification.	See note 1	X
5.2.10.1	1.3.10.1		Periodic broadcast reservation burst format.	See note 1	X
5.2.10.1.1	1.3.10.1	a		Periodic_NonDitherRes Periodic_DitherRes	E
5.2.10.1.2	1.3.10.1	b		See note 1a	X
5.2.10.1.3	1.3.10.1.2	a		See note 1a	X
5.2.10.1.4	1.3.10.1.2	b		Periodic_DitherRes	E
5.2.10.1.5	1.3.10.1.2	c		Periodic_NonDitherRes	E
5.2.10.2	1.3.10.2		Periodic broadcast timers.	See note 1	X
5.2.10.2.1	1.3.10.2.1	a		See note 2	O
5.2.10.2.2	1.3.10.2.1	b		Periodic_IndependentStreams	E
5.2.10.3	1.3.10.3		Periodic broadcast parameters.	See note 1	X
5.2.10.3.1	1.3.10.3	a		See note 2	O
5.2.10.3.2	1.3.10.3	b		See note 2	O
5.2.10.3.3	1.3.10.3	c		See note 2	O
5.2.10.3.4	1.3.10.3.1	a		Periodic_TV11	E
5.2.10.3.5	1.3.10.3.2	a		Periodic_Rate	E
5.2.10.3.6	1.3.10.3.3	a		Periodic_DitherRange	E
5.2.10.3.7	1.3.10.3.3	b		Periodic_DitherRange	E
5.2.10.4	1.3.10.4		Periodic broadcast reception procedures.	See note 1	X
5.2.10.4.1	1.3.10.4.1	a		Periodic_NonDitherRes Periodic_DitherRes Periodic_Cancel	E
5.2.10.4.2	1.3.10.4.1	b		See note 1a	X
5.2.10.4.3	1.3.10.4.2	a		Periodic_Replacement	E
5.2.10.4.4	1.3.10.4.3	a		Periodic_CancelIncremental Periodic_CancelUnicast	E
5.2.10.5	1.3.10.5		Periodic broadcast transmission procedures.	See note 1	X
5.2.10.5.1	1.3.10.5.1.1	a		Periodic_Rate Sync_Interval	E
5.2.10.5.2	1.3.10.5.1.2	a		See note 2	O
5.2.10.5.3	1.3.10.5.2	a		Periodic_Rate	E
5.2.10.5.4	1.3.10.5.2	b		Periodic_DitherRange	E
5.2.10.5.5	1.3.10.5.2	c		See note 2	O
5.2.10.5.6	1.3.10.5.2	d		See note 2	O
5.2.10.5.7	1.3.10.5.2	e		See note 2	O
5.2.10.5.8	1.3.10.5.2	f		See note 2	O
5.2.10.5.9	1.3.10.5.3	a		See note 2	O
5.2.10.5.10	1.3.10.5.3	b		See note 1a	X
5.2.10.5.11	1.3.10.5.3	c		Periodic_Availability_A Periodic_Availability_B	E
5.2.10.5.12	1.3.10.5.3	d		Periodic_Availability_A	E
5.2.10.5.13	1.3.10.5.3	e		Periodic_Availability_B	E
5.2.10.5.14	1.3.10.5.4	a		Periodic_TV11 Periodic_Availability_A	E
5.2.10.5.15	1.3.10.5.5	a		Periodic_InitialRes	E
5.2.10.5.16	1.3.10.5.5	b		Periodic_InitialRes	E
5.2.10.5.17	1.3.10.5.6	a		Periodic_DitherOffset_A	E

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.2.10.5.18	1.3.10.5.6	b		Periodic_DitherOffset_B	E
5.2.10.5.19	1.3.10.5.6	c		Periodic_DitherRange Periodic_DitherOffset_C	E
5.2.10.5.20	1.3.10.5.7.1	a		Periodic_DitherOffset_B	E
5.2.10.5.21	1.3.10.5.7.2	a		See note 2	O
5.2.10.5.22	1.3.10.5.7.3	a		Periodic_InitialRes	E
5.2.10.5.23	1.3.10.5.8.1	a		See note 2	O
5.2.10.5.24	1.3.10.5.8.1	b		Periodic_DitherOffset_D Periodic_Availability_A	E
5.2.10.5.25	1.3.10.5.8.2	a		Periodic_Availability_A Periodic_Availability_B	E
5.2.10.5.26	1.3.10.5.8.3	a		See note 2	O
5.2.10.5.27	1.3.10.5.9	a		See note 2	O
5.2.10.5.28	1.3.10.5.9	b		Periodic_Cancel	E
5.2.11	1.3.11		Incremental broadcast protocol specification.	See note 1	X
5.2.11.1	1.3.11.1		Incremental broadcast reservation burst format.	See note 1	X
5.2.11.1.1	1.3.11.1	a		Incremental_Reservation_A	E
5.2.11.1.2	1.3.11.1	b		See note 1a	X
5.2.11.1.3	1.3.11.1.2	a		See note 1a	X
5.2.11.1.4	1.3.11.1.2	b		Incremental_Reservation_A	E
5.2.11.2	1.3.11.2		Incremental broadcast parameters.	See note 1	X
5.2.11.2.1	1.3.11.2	a		See note 2	O
5.2.11.2.2	1.3.11.2	b		See note 2	O
5.2.11.2.3	1.3.11.2.1	a		Incremental_Request	E
5.2.11.2.4	1.3.11.2.2	a		Incremental_Request	E
5.2.11.2.5	1.3.11.2.2	b		Incremental_Request	E
5.2.11.3	1.3.11.3		Incremental broadcast reception procedures.	See note 1	X
5.2.11.3.1	1.3.11.3.1	a		Incremental_Reservation_A	E
5.2.11.3.2	1.3.11.3.2	a		Incremental_Reservation_B	E
5.2.11.4	1.3.11.4		Incremental broadcast transmission procedures.	See note 1	X
5.2.11.4.1	1.3.11.4.1	a		See note 2	O
5.2.11.4.2	1.3.11.4.1	b		See note 1a	X
5.2.11.4.3	1.3.11.4.2	a		Incremental_SlotSel	E
5.2.11.4.4	1.3.11.4.2	b		See note 1a	X
5.2.11.4.5	1.3.11.4.3	a		Incremental_Request	E
5.2.12	1.3.12		Combined periodic broadcast and incremental broadcast protocol specification.	See note 1	X
5.2.12.1	1.3.12.1		Combined periodic broadcast and incremental broadcast reservation burst.	See note 1	X
5.2.12.1.1	1.3.12.1	a		Combined_Reservation NetEntry_Periodic	E
5.2.12.1.2	1.3.12.1	b		See note 1a	X
5.2.12.1.3	1.3.12.1	c		See note 1a	X
5.2.12.1.4	1.3.12.1	d		See note 1a	X
5.2.12.1.5	1.3.12.1	e		Combined_Reservation	E
5.2.13	1.3.13		Big negative dither (BND) broadcast protocol specifications.	See note 1	X
5.2.13.1	1.3.13.1		BND reservation burst format	See note 1	X
5.2.13.1.1	1.3.13.1	a		BND_Reservation	E
5.2.13.1.2	1.3.13.1	b		See note 1a	X
5.2.13.2	1.3.13.2		BND broadcast parameters.	See note 1	X
5.2.13.2.1	1.3.13.2	a		See note 1a	X
5.2.13.3	1.3.13.3		BND broadcast reception procedures.	See note 1	X
5.2.13.3.1	1.3.13.3	a		BND_Reservation	E
5.2.13.4	1.3.13.4		BND broadcast transmission procedures	See note 1	X
5.2.13.4.1	1.3.13.4	a		See note 2	X
5.2.14	1.3.14		Unicast request protocol specification.	See note 1	X
5.2.14.1	1.3.14.1		Unicast request reservation burst format.	See note 1	X
5.2.14.1.1	1.3.14.1.1	a		Unicast_Reservation_A	E
5.2.14.1.2	1.3.14.1.2	a		See note 1a	X
5.2.14.1.3	1.3.14.1.2	b		See note 1a	X
5.2.14.1.4	1.3.14.1.2	c		See note 1a	X

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.2.14.1a	1.3.14.2		Unicast request parameters	See note 1	X
5.2.14.1a.1	1.3.14.2	a		See note 1a	X
5.2.14.1a.2	1.3.14.2	b		See note 2	O
5.2.14.1a.3	1.3.14.2.1	a		See note 1a	X
5.2.14.1a.4	1.3.14.2.2	a		See note 1a	X
5.2.14.1a.5	1.3.14.2.3	a		See note 1a	X
5.2.14.1a.6	1.3.14.2.3	b		See note 1a	X
5.2.14.1a.7	1.3.14.2.4	a		See note 1a	X
5.2.14.1a.8	1.3.14.2.4	b		See note 1a	X
5.2.14.1a.9	1.3.14.2.4	c		See note 1a	X
5.2.14.1a.10	1.3.14.2.5	a		See note 1a	X
5.2.14.2	1.3.14.3		Unicast request reception procedures.	See note 1	X
5.2.14.2.1	1.3.14.3	a		Unicast_Reservation_A Unicast_Reservation_B Unicast_Reservation_C	E
5.2.14.3	1.3.14.4		Unicast request transmission procedures	See note 1	X
5.2.14.3.1	1.3.14.4.1	a		See note 2	O
5.2.14.3.2	1.3.14.4.1	b		See note 1a	X
5.2.14.3.3	1.3.14.4.2	a		See note 2	O
5.2.14.3.4	1.3.14.4.2	b		See note 1a	X
5.2.14.3.5	1.3.14.4.3.1	a		See note 1a	X
5.2.14.3.6	1.3.14.4.3.1	b		See note 1a	X
5.2.14.3.7	1.3.14.4.3.2	a		See note 1a	X
5.2.14.3.8	1.3.14.4.3.2	b		See note 1a	X
5.2.14.3.9	1.3.14.4.4	a		See note 2	O
5.2.14.3.10	1.3.14.4.5	a		Unicast_Reservation_D	E
5.2.15	1.3.15		Information transfer request protocol specification.	See note 1	X
5.2.15.1	1.3.15.1		Information transfer request reservation burst format.	See note 1	X
5.2.15.1.1	1.3.15.1	a		Info_Reservation	E
5.2.15.1.2	1.3.15.1	b		See note 1a	X
5.2.15.1.3	1.3.15.1	c		See note 1a	X
5.2.15.2	1.3.15.3		Information transfer request reception procedures.	See note 1	X
5.2.15.2.1	1.3.15.3	a		Info_Reservation	E
5.2.15.2.2	1.3.15.3	b		Info_Reservation	E
5.2.16	1.3.16		Directed request protocol specification.	See note 1	X
5.2.16.1	1.3.16.1		Directed request reservation burst format.	See note 1	X
5.2.16.1.1	1.3.16.1	a		Autotune_Reservation	E
5.2.16.1.2	1.3.16.1	b		See note 2	O
5.2.16.1.3	1.3.16.1	c		See note 1a	X
5.2.16.1.4	1.3.16.1	d		See note 1a	X
5.2.16.1.5	1.3.16.1	e		See note 1a	X
5.2.16.1.6	1.3.16.1	f		See note 1a	X
5.2.16.1.7	1.3.16.1.1	a		See note 1a	X
5.2.16.1.8	1.3.16.1.1	b		See note 1a	X
5.2.16.1.9	1.3.16.1.1	c		Autotune_Invalid_A	E
5.2.16.1.10	1.3.16.1.2	a		PleaResponse_Reservation_A PleaResponse_Reservation_B	E
5.2.16.1.11	1.3.16.1.2	b		PleaResponse_Reservation_A PleaResponse_Reservation_B	E
5.2.16.1.12	1.3.16.1.2	c		PleaResponse_Reservation_A PleaResponse_Reservation_B	E
5.2.16.1.13	1.3.16.1.2	d		See note 1a	X
5.2.16.1.14	1.3.16.1.2	e		See note 1a	X
5.2.16.2	1.3.16.2		Directed request parameters.	See note 1	X
5.2.16.2.1	1.3.16.2	a		See note 2	O
5.2.16.2.2	1.3.16.2	b		See note 2	O
5.2.16.2.3	1.3.16.2.1	a		See note 1a	X
5.2.16.3	1.3.16.3		Directed request reception procedures.	See note 1	X
5.2.16.3.1	1.3.16.3.1	a		Autotune_Reservation Autotune_CancelAbsent	E

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.2.16.3.2	1.3.16.3.1.2	a		Autotune_Override	E
5.2.16.3.3	1.3.16.3.1.2	b		Autotune_CancelAbsent	E
5.2.16.3.4	1.3.16.3.1.3	a		Autotune_Invalid_B	E
5.2.16.3.5	1.3.16.3.2.1	a		PleaResponse_Reservation_A PleaResponse_Reservation_B	E
5.2.16.3.6	1.3.16.3.2.2	a		PleaResponse_Reservation_A	E
5.2.16.3.7	1.3.16.3.2.3	a		PleaResponse_Reservation_B	E
5.2.16.4	1.3.16.4		Directed request transmission procedures.	See note 1	X
5.2.16.4.1	1.3.16.4.1.1	a		See note 4	O
5.2.16.4.2	1.3.16.4.1.1	b		See note 4	O
5.2.16.4.3	1.3.16.4.1.2	a		See note 4	O
5.2.16.4.4	1.3.16.4.2	a		See note 4	O
5.2.16.4.5	1.3.16.4.2	b		See note 4	O
5.2.16.4.6	1.3.16.4.3	a		PleaResponse_Retransmission	E
5.2.16.4.7	1.3.16.4.3	b		See note 4	O
5.2.16.4.8				See note 4	O
5.2.16.4.9	1.3.16.4.4	a		See note 4	O
5.2.16.4.10	1.3.16.4.4	b		See note 4	O
5.2.16.4.11	1.3.16.4.5.1	a		PleaResponse_Transmission_A	E
5.2.16.4.12	1.3.16.4.5.1	b		PleaResponse_Transmission_A	E
5.2.16.4.13	1.3.16.4.5.2	a		See note 2	O
5.2.16.4.14	1.3.16.4.5.3	a		PleaResponse_Transmission_B	E
5.2.16.4.15	1.3.16.4.5.3	b		PleaResponse_Transmission_B	E
5.2.16.5	1.3.16.5		Directed request response procedures	See note 1	X
5.2.16.5.1	1.3.16.5.1	a		Autotune_Directed_A Autotune_Directed_B	E
5.2.16.5.2	1.3.16.5.1	b		Autotune_Directed_A Autotune_Directed_B	E
5.2.16.5.3	1.3.16.5.1	c		Autotune_Directed_A Autotune_Directed_B	E
5.2.16.5.4	1.3.16.5.1	d		Autotune_Directed_A Autotune_Directed_B	E
5.2.16.5.5	1.3.16.5.1	e		Autotune_Directed_A Autotune_Directed_B	E
5.2.16.5.6	1.3.16.5.1	f		Autotune_DirectedCessation	E
5.2.16.5.7	1.3.16.5.1.1	a		See note 4	O
5.2.16.5.8	1.3.16.5.1.1	b		See note 2	O
5.2.16.5.9	1.3.16.5.2	a		Autotune_Autonomous_A Autotune_Autonomous_B Autotune_Autonomous_C Autotune_Autonomous_D Autotune_Autonomous_E	E
5.2.16.5.10	1.3.16.5.2	b		Autotune_AutonomousCessation	E
5.2.16.5.11	1.3.16.5.3.1	a		Autotune_DirectedCancelAutonom ous Autotune_Autonomous_A Autotune_Autonomous_B	E
5.2.16.5.12	1.3.16.5.3.2	a		Autotune_Autonomous_C	E
5.2.16.5.13	1.3.16.5.4	a		Autotune_CancelResp	E
5.2.16.5.14	1.3.16.5.5	a		Autotune_Unsupported	E
5.2.16.5.15	1.3.16.5.5	b		Autotune_Unsupported	E
5.2.16.5.16	1.3.16.5.5	c		Autotune_Unsupported	E
5.2.16.5.17	1.3.16.5.6	a		Autotune_Unsupported	E
5.2.16.5.18	1.3.16.5.6	b		See note 4	O
5.2.17	1.3.17		Block reservation protocols specification.	See note 1	X
5.2.17.1	1.3.17.1		Superframe block reservation burst format.	See note 1	X
5.2.17.1.1	1.3.17.1	a		Block_Superframe_Reserve_A	E
5.2.17.1.2	1.3.17.1	b		See note 1a	X
5.2.17.1.3	1.3.17.1	c		See note 1	X
5.2.17.2	1.3.17.2		Second frame block reservation burst format.	See note 1	X
5.2.17.2.1	1.3.17.2	a		Block_Secondframe_Reserve	E
5.2.17.2.2	1.3.17.2	b		See note 1a	X
5.2.17.4	1.3.17.4		Superframe block reservation reception procedures.	See note 1	X

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.2.17.4.2	1.3.17.4.1	a		Block_Superframe_Reserve_A Block_Superframe_Reserve_B Block_Superframe_Reserve_C Block_Superframe_Invalid	E
5.2.17.4.3	1.3.17.4.2	a		Block_Superframe_Reserve_A Block_Superframe_Reserve_B Block_Superframe_Reserve_C Block_Superframe_Invalid	E
5.2.17.5	1.3.17.5		Second frame block reservation parameters.	See note 1	X
5.2.17.5.1	1.3.17.5	a		See note 4	O
5.2.17.5.2	1.3.17.5	b		See note 4	O
5.2.17.5.3	1.3.17.5.1	a		Block_SecondFrame_Reserve	E
5.2.17.5.3a	1.3.17.5.1	b		Block_SecondFrame_Reserve	E
5.2.17.5.4	1.3.17.5.2	a		See note 4	O
5.2.17.6	1.3.17.6		Second frame block reservation reception procedures.	See note 1	X
5.2.17.6.1		a		Block_SecondFrame_Reserve	E
5.2.17.7a	1.3.17.8		Procedures for responding to a rebroadcast request for superframe block reservation.	See note 1	X
5.2.17.7a.1	1.3.17.8	a		Block_Superframe_ReBroadcast	E
5.2.17.7a.2	1.3.17.8	b		Block_Superframe_ReBroadcast	E
5.2.17.7a.3	1.3.17.8	c		Block_Superframe_ReBroadcast	E
5.2.17.7a.4	1.3.17.8	d		Block_Superframe_ReBroadcast	E
5.2.18	1.3.18		Response protocol specification.	See note 1	X
5.2.18.1	1.3.18.1		Response burst format.	See note 1	X
5.2.18.1.1	1.3.18.1	a		Response_Reservation	E
5.2.18.1.2	1.3.18.1	b		Response_Reservation	E
5.2.18.1.3	1.3.18.1	c		See note 1a	X
5.2.18.1.4	1.3.18.1	e		See note 2	O
5.2.18.1.5	1.3.18.1	d		See note 1a	X
5.2.18.1.6	1.3.18.1	f		Response_Reservation	E
5.2.18.1.7	1.3.18.1	g		See note 2	O
5.2.19	1.3.19		General request protocol specification.	See note 1	X
5.2.19.1	1.3.19.1		General request burst format.	See note 1	X
5.2.19.1.1	1.3.19.1.1	a		See note 4	O
5.2.19.1.2	1.3.19.1.1	b		See note 4	O
5.2.19.1.3	1.3.19.1.1	c		See note 2	O
5.2.19.1.4	1.3.19.1.2	a		See note 1a	X
5.2.19.1.5	1.3.19.1.3	a		See note 2	O
5.2.19.2	1.3.19.2		General request procedures.	See note 1	X
5.2.19.2.1	1.3.19.2.1	a		See note 2	O
5.2.19.2.4	1.3.19.2.2	a		See note 2	O
5.2.19.2.5	1.3.19.2.3.1	a		Request_Unsupported	E
5.2.19.2.6	1.3.19.2.3.2	a		See note 2	O
5.2.20	1.3.20		General response protocol specification.	See note 1	X
5.2.20.1	1.3.20.1		General response burst format.	See note 1	X
5.2.20.1.1	1.3.20.1.1	a		Request_Unsupported	E
5.2.20.1.2	1.3.20.1.1	b		Request_Unsupported	E
5.2.20.1.3	1.3.20.1.1	c		Request_Unsupported	E
5.2.20.1.4	1.3.20.1.1	d		Request_Unsupported	E
5.2.20.1.5	1.3.20.1.3	a		See note 2	O
5.2.20.1.6	1.3.20.1.3	b		See note 2	O
5.2.20.1.7	1.3.20.1.3	c		Request_Unsupported	E
5.2.20.1.8	1.3.20.1.4	a		See note 1a	X
5.2.20.1.9	1.3.20.1	j		See note 1a	X
5.2.20.1.10	1.3.20.1	k		See note 1a	X
5.2.20.2	1.3.20.2		General response procedures.	See note 1	X
5.2.20.2.1	1.3.20.2.1	a		See note 2	O
5.2.20.2.2	1.3.20.2.2	a		See note 2	O
5.2.21	1.3.21		Retransmission procedures	See note 1	X
5.2.21.1	1.3.21.1	a		See note 2	O
5.2.21.2	1.3.21.2	a		See note 1a	X
5.3	1.4		DLS sublayer	See note 1	X
5.3.1	1.4.1		Services	See note 1	X

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.3.1.1			General	See note 1	X
5.3.1.1.1	1.4.1.1.2	a		See note 1a	X
5.3.1.2	1.4.1.2		Data transfer	See note 1	X
5.3.1.2.1	1.4.1.2	a		See note 1a	X
5.3.1.2.2	1.4.1.2	b		See note 1a	X
5.3.1.3	1.4.2.2		Station address encoding	See note 1	X
5.3.1.3.1	1.4.2.2.1	a		See note 1a	X
5.3.1.3.2a	1.4.2.2.2.1	a		See note 2	O
5.3.1.3.2b	1.4.2.2.2.1	b		See note 2	O
5.3.1.3.2c	1.4.2.2.2.1	c		See note 2	O
5.3.1.3.2d	1.4.2.2.2.1	d		See note 2	O
5.3.1.3.2e	1.4.2.2.2.2	a		See note 2	O
5.3.1.3.2f	1.4.2.2.2.2	b		See note 2	O
5.3.1.3.2g	1.4.2.2.2.2	c		See note 2	O
5.3.1.3.3	1.4.2.2.2.3	a		See note 1a	X
5.3.1.3.4	1.4.2.2.3	a		See note 1a	X
5.3.1.3.8	1.4.2.2.6	a		See note 2	O
5.3.1.3.9	1.4.2.2.6.1	a		See note 1a	X
5.3.1.4	1.4.2.3		DLS burst formats	See note 1	X
5.3.1.4.1	1.4.2.3.10.1	a		See note 2	O
5.3.1.4.2	1.4.2.3.10.1	b		See note 1a	X
5.3.1.4.3	1.4.2.3.10.2	a		See note 1a	X
5.3.1.4.4	1.4.2.3.10.2	b		See note 1a	X
5.3.2	1.4.3		DLS system parameters	See note 1	X
5.3.2.1	1.4.3	a		See note 1a	X
5.3.2.1	1.4.3.6		Parameter ND4 (maximum length of a UDATA burst)	See note 1	X
5.3.2.1.1	1.4.3.6	a		See note 1a	X
5.3.3	1.4.4		DLS Procedures	See note 1	X
5.3.3.1	1.4.4.1		Broadcast	See note 1	X
5.3.3.1.1	1.4.4.1	a		See note 2	O
5.3.3.1.2	1.4.4.5.4	a		DLS_UDATA_Receive	E
5.3.3.2	1.4.4.7		DLS not supported	See note 1	X
5.3.3.2.1	1.4.4.7	a		DLS_NotSupported	E
5.3.3.3	1.4.4.9		User data packet reception	See note 1	X
5.3.3.3.1	1.4.4.9.1.2	a		DLS_UDATA_Receive	E
5.3.3.3.2	1.4.4.9.1.2	b		See note 2	O
5.3.3.3.3	1.4.4.9.3	a		DLS_UDATA_Receive	E
5.3.3.4	1.4.4.13		CTRL DLPDU	See note 1	X
5.3.3.4.1	1.4.4.13	a		See note 1a	X
5.4	1.5		Link Management Entity sublayer.	See note 1	X
5.4.1	1.5.1		Services.	See note 1	X
5.4.1.1				See note 4	O
5.4.2	1.5.2		Synchronization burst format.	See note 1	X
5.4.2.1			General	See note 1	X
5.4.2.1.1	1.5.2	a		Sync_Format Sync_Format_Rec	E
5.4.2.2	1.5.2.1		Fixed and variable data fields.	See note 1	X
5.4.2.2.1	1.5.2.1	a		See note 1a	X
5.4.2.3	1.5.2.2		Fixed data field format.	See note 1	X
5.4.2.3.1	1.5.2.2.1	a		Sync_Format Sync_Format_Rec	E
5.4.2.3.2	1.5.2.2.2	a		Sync_Fixed_NIC Sync_Fixed_BaseAlt Sync_Fixed_DataAge CPR_Encode CPR_Decode	E
5.4.2.3.3	1.5.2.2.2	b		See note 1a	X
5.4.2.3.4	1.5.2.2.2	c		See note 1a	X
5.4.2.3.5	1.5.2.2.2	d		See note 1a	X
5.4.2.3.6	1.5.2.2.2	ef		See note 1a	X
5.4.2.3.7	1.5.2.2.2	f		See note 1a	X
5.4.2.3.8	1.5.2.2.2	g		See note 1a	X
5.4.2.3.9	1.5.2.2.2	h		See note 1a	X

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.4.2.3.10	1.5.2.2.3	a		Sync_Fixed_NIC	E
5.4.2.3.11	1.5.2.2.4	a		Sync_Fixed_BaseAlt	E
5.4.2.3.12	1.5.2.2.5	a		Sync_Fixed_DataAge	E
5.4.2.3.13	1.5.2.2.5	b		Sync_Fixed_NIC	E
5.4.2.4	1.5.2.3		Variable data field format.	See note 1	X
5.4.2.4.1	1.5.2.3	a		See note 2	O
5.4.2.4.2	1.5.2.3	b		See note 2	O
5.4.2.4.3	1.5.2.3	c		See note 2	O
5.4.2.5	1.5.2.4		Synchronization burst request.	See note 1	X
5.4.2.5.1	1.5.2.4	a		See note 2	O
5.4.2.6	1.5.2.6		Broadcast link management burst	See note 1	X
5.4.2.6.1	1.5.2.6.1	a		See note 4	O
5.4.2.6.2	1.5.2.6.2	a		See note 1a	X
5.4.2.6.3	1.5.2.6.2	b		See note 1a	X
5.4.2.6.4	1.5.2.6.2	c		See note 1a	X
5.4.3	1.5.3		Control (CTRL) parameter formats.	See note 1	X
5.4.3.1	1.5.3.1		Encoding	See note 1	X
5.4.3.1.1	1.5.3.1	a		See note 1a	X
5.4.3.1.2	1.5.3.1	b		See note 1a	X
5.4.3.2	1.5.3.2		VDL Mode 4 parameter identification	See note 1	X
5.4.3.2.1	1.5.3.2	a		See note 1a	X
5.4.3.3	1.5.3.5		Ground-initiated modification parameters	See note 1	X
5.4.3.3.1	1.5.3.5	a		UCTRL_param_VS1 UCTRL_param_VS2 UCTRL_param_VS4 UCTRL_param_Q4 UCTRL_param_CG1_limit UCTRL_param_CG1_range UCTRL_param_CG1_inc UCTRL_param_CG1_decay UCTRL_param_M2inc UCTRL_param_M2limit	E
5.4.3.3.1a				See note 2	O
5.4.3.3.1b	1.5.3.5	b		See note 1a	X
5.4.3.3.2	1.5.3.5.2	a		UCTRL_param_VS1 UCTRL_param_VS2 UCTRL_param_VS4	E
5.4.3.3.3	1.5.3.5.3	a		UCTRL_param_Q4	E
5.4.3.3.4	1.5.3.5.4	a		See note 1a	X
5.4.3.3.5	1.5.3.5.4	b		UCTRL_param_M2inc	E
5.4.3.3.6	1.5.3.5.4	c		UCTRL_param_M2limit	E
5.4.3.3.7	1.5.3.5.5	a		See note 1a	X
5.4.3.3.8	1.5.3.5.5	b		See note 2	O
5.4.3.3.9	1.5.3.5.5	c		UCTRL_param_CG1_range	E
5.4.3.3.10	1.5.3.5.5	d		See note 2	O
5.4.3.3.11	1.5.3.5.5	e		UCTRL_param_CG1_inc	E
5.4.3.3.12	1.5.3.5.5	f		UCTRL_param_CG1_decay	E
5.4.3.3.13	1.5.3.5.5	g		UCTRL_param_CG1_limit	E
5.4.3.3.13a	1.5.3.5.7	a		see note 1a	X
5.4.3.3.13b	1.5.3.5.7	b		see note 1a	X
5.4.3.3.13c	1.5.3.5.7	c		see note 1a	X
5.4.3.3.13d	1.5.3.5.7	d		see note 1a	X
5.4.3.3.14	1.5.3.5.9	a		See note 1a	X
5.4.3.4	1.5.3.6		Ground-initiated information parameters	See note 1	X
5.4.3.4.1	1.5.3.6	a		See note 1a	X
5.4.3.4.1a	1.5.3.6	b		See note 1a	X
5.4.3.4.2	1.5.3.6.6	a		See note 1a	X
5.4.3.4.3	1.5.3.6.6.2	a		See note 1a	X
5.4.3.4.4	1.5.3.6.6.3	a		See note 1a	X
5.4.3.4.5	1.5.3.6.6.3	b		See note 1a	X
5.4.3.4.6	1.5.3.6.6.3	c		See note 1a	X
5.4.3.4.7	1.5.3.6.6.3	d		See note 1a	X
5.4.3.4.8	1.5.3.6.6.3	e		See note 2	O
5.4.3.4.9	1.5.3.6.6.4	a		See note 2	O

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.4.3.4.10	1.5.3.6.6.5	a		See note 2	O
5.4.3.4.11	1.5.3.6.6.6	a		See note 2	O
5.4.3.4.12	1.5.3.6.6.7	a		See note 2	O
5.4.3a	1.5.4		LME timers and parameters	see note 1	X
5.4.3a.1			General	see note 1	X
5.4.3a.1.1	1.5.4	a		see note 2	O
5.4.3a.2	1.5.4.1		Counter L1 (maximum number of missed reservations) and Timer TL3 (inter-miss timer)	see note 1	X
5.4.3a.2.1	1.5.4.1	a		see note 1a	X
5.4.3a.2.2	1.5.4.1	b		Param_L1	E
5.4.3a.2.3	1.5.4.1	c		Param_L1	E
5.4.3a.2.4	1.5.4.1	d		Param_L1	E
5.4.3a.2.5	1.5.4.1	e, f		Param_L1	E
5.4.3a.2.6	1.5.4.1	g		see note 2	O
5.4.3a.2.7	1.5.4.1	h		see note 2	O
5.4.4	1.5.5		LME procedures.	See note 1	X
5.4.4.1	1.5.5.1		Synchronization burst procedures.	See note 1	X
5.4.4.1.1	1.5.5.1	a		Sync_Format	E
5.4.4.1.2	1.5.5.1	b		See note 2	O
5.4.4.1.3	1.5.5.1	c		See note 2	O
5.4.4.1.4	1.5.5.1	d		Sync_Latency	E
5.4.4.1.5	1.5.5.1	e		See note 2	O
5.4.4.1.6	1.5.5.1.1.2	a		See note 2	O
5.4.4.1.6a	1.5.5.1.1			See note 2	O
5.4.4.1.6b	1.5.5.1.1			See note 2	O
5.4.4.1.6c	1.5.5.1.2	a		Sync_Rate	E
5.4.4.1.6d	1.5.5.1.2	b		See note 2	O
5.4.4.1.7	1.5.5.1.3.2	a		See note 2	O
5.4.4.1.8	1.5.5.1.4	a		Conflict_Periodic_B Conflict_NoAction	E
5.4.4.1.9	1.5.5.1.4	b		Conflict_Periodic_B Conflict_NoAction	E
5.4.4.2	1.5.5.2		Peer entity contact Table (PECT)	See note 1	X
5.4.4.2.1	1.5.5.2	a		See note 2	O
5.4.4.2.2	1.5.5.2	b		See note 2	O
5.4.4.2.3	1.5.5.2	c		See note 2	O
5.4.4.2.4				See note 2	O
5.4.4.2.5				See note 2	O
5.4.4.2.6				See note 2	O
5.4.4.2.7				See note 2	O
5.4.4.2.8				See note 4	O
5.4.4.2.9				See note 2	O
5.4.4.2.10				See note 2	O
5.4.4.2.11				See note 2	O
5.4.4.2.12				See note 2	O
5.4.4.2.13				See note 2	O
5.4.4.3	1.5.5.3		Network entry protocol specifications.	See note 1	X
5.4.4.3.a	1.5.5.3.1	a		See note 2	O
5.4.4.3.b	1.5.5.3.1.1	a		See note 2	O
5.4.4.3.c	1.5.5.3.1.1	b		See note 2	O
5.4.4.3.d	1.5.5.3.1.2	a		UCTRL_param_CG1_range	E
5.4.4.3.1	1.5.5.3.1.3	a		See note 2	O
5.4.4.3.2	1.5.5.3.1.3	b		See note 2	O
5.4.4.3.3	1.5.5.3.1.3	c		See note 2	O
5.4.4.3.3a	1.5.5.3.1.4	a		UCTRL_param_CG1_limit	E
5.4.4.3.3b	1.5.5.3.1.4	b		UCTRL_param_CG1_limit	E
5.4.4.3.3c	1.5.5.3.1.5	a		UCTRL_param_CG1_inc	E
5.4.4.3.3d	1.5.5.3.1.6	a		UCTRL_param_CG1_decay	E
5.4.4.3.3e	1.5.5.3.1.7	a		See note 4	O
5.4.4.3.3f	1.5.5.3.1.8	a		Param_CG1_reach	E
5.4.4.3.3g	1.5.5.3.1.9.1	a		See note 2	O
5.4.4.3.3h	1.5.5.3.1.9.1	b		See note 2	O
5.4.4.3.3i	1.5.5.3.1.9.2	a		See note 2	O

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.4.4.3.3j	1.5.5.3.1.9.2	b		See note 2	O
5.4.4.3.4a	1.5.5.3.2.1	a		NetEntry_OneMinute NetEntry_Periodic NetEntry_Delayed_BND NetEntry_Delayed_Plea UCTRL_param_M2limit UCTRL_param_CG1_limit	E
5.4.4.3.4b	1.5.5.3.2.2	a		UCTRL_param_CG1_inc UCTRL_param_CG1_range UCTRL_param_CG1_decay	E
5.4.4.3.4c	1.5.5.3.2.3	a		UCTRL_param_CG1_limit	E
5.4.4.3.4d				See note 2	O
5.4.4.3.4e				See note 4	O
5.4.4.3.4f				See note 2	O
5.4.4.3.4g	1.5.5.3.2.4	a		NetEntry_Repeat	E
5.4.4.3.4h	1.5.5.3.3.1.1	a		NetEntry_Delayed_Plea	E
5.4.4.3.4i	1.5.5.3.3.1.1	b		See note 1a	X
5.4.4.3.4j	1.5.5.3.3.1.1	c		See note 2	O
5.4.4.3.4k	1.5.5.3.3.1.1	d		See note 2	O
5.4.4.3.4l	1.5.5.3.3.1.1	e		NetEntry_Delayed_Plea	E
5.4.4.3.4m	1.5.5.3.3.1.1	f		NetEntry_Delayed_Plea	E
5.4.4.3.4n	1.5.5.3.3.1.1	g		See note 2	O
5.4.4.3.4o	1.5.5.3.3.1.2	a		See note 2	O
5.4.4.3.4p	1.5.5.3.3.1.3	a		See note 2	O
5.4.4.3.4q	1.5.5.3.3.1.3	b		See note 2	O
5.4.4.3.4r	1.5.5.3.3.1.3	c		See note 2	O
5.4.4.3.5	1.5.5.3.3.2	a		See note 2	O
5.4.4.3.6	1.5.5.3.3.2	b		NetEntry_Receive	E
5.4.4.3.7	1.5.5.3.3.2	c		See note 2	O
5.4.4.3.8	1.5.5.3.3.2	d		See note 2	O
5.4.4.3.9	1.5.5.3.3.2	e		See note 2	O
5.4.4.3.10	1.5.5.3.3.3	a		See note 4	O
5.4.4.3.11	1.5.5.3.3.3	b		See note 4	O
5.4.4.3.12	1.5.5.3.3.3	d		See note 4	O
5.4.4.3.12a	1.5.5.3.3.4	a		See note 2	O
5.4.4.3.12b	1.5.5.3.4.1	a		NetEntry_Delayed_BND	E
5.4.4.3.12c	1.5.5.3.4.1	b		See note 2	O
5.4.4.3.12d	1.5.5.3.4.1	c		See note 2	O
5.4.4.3.12e	1.5.5.3.4.1	d		See note 2	O
5.4.4.3.12f	1.5.5.3.4.1	e		See note 2	O
5.4.4.3.12g	1.5.5.3.4.2	a		See note 2	O
5.4.4.3.12h	1.5.5.3.4.3	a		See note 2	O
5.4.4.3.12i	1.5.5.3.4.3	b		See note 2	O
5.4.4.3.13	1.5.5.3.5	a		NetEntry_OneMinute NetEntry_Periodic	E
5.4.5	3		Additional material for ADS-B applications.	See note 2	O
5.4.5.1	3.3		Information field formats	See note 1	X
5.4.5.1.1	3.3.1	a		See note 1a	X
5.4.5.2	3.4.1		ADS-B request format	See note 1	X
5.4.5.2.1	3.4.1	a		See note 7	X
5.4.5.2.2	3.4.1	b		See note 7	X
5.6	4		Definitions For Compact Position Reporting	See note 1	X
5.6.1	4.1		Introduction	See note 1	X
5.6.2	4.2		Parameter symbols, data types, constants and variables	See note 1	X
5.6.2.1	4.2.1		Parameter symbols	See note 1	X
5.6.2.2	4.2.2		Data types	See note 1	X
5.6.2.2.1	4.2.2.1	a		See note 1a	X
5.6.2.2.2	4.2.2.2	a		See note 1a	X
5.6.2.3	4.2.3		Constants	See note 1	X
5.6.2.3.1	4.2.3	a		See note 1a	X
5.6.2.4	4.2.4		Variables	See note 1	X
5.6.2.4.1	4.2.4	a		See note 1a	X

Requirement reference	Reference in [1]	Req	Title	Test Case	E/O/X
5.6.2.4.2	4.2.4	b		See note 1a	X
5.6.2.4.3	4.2.4	c		See note 1a	X
5.6.2.5	4.2.5		Functions	See note 1	X
5.6.2.5.1	4.2.5	a		See note 1a	X
5.6.2.6	4.2.6		Patch constants	See note 1	X
5.6.2.6.1	4.2.6.1	a		See note 1a	X
5.6.2.6.2	4.2.6.2	a		See note 1a	X
5.6.3	4.3		Fixed Data Field Position Encoding	See note 1	X
5.6.3.1	4.3.1		General	See note 1	X
5.6.3.1.1	4.3.1	a		CPR_Encode	E
5.6.3.2	4.3.2		Input parameters	See note 1	X
5.6.3.2.1	4.3.2	a		See note 1a	X
5.6.3.3	4.3.3		Calculations	See note 1	X
5.6.3.3.1	4.3.3.1	a		CPR_Encode	E
5.6.3.3.2	4.3.3.2	a		CPR_Encode	E
5.6.4	4.4		Fixed Data Field Position Local Decoding	See note 1	X
5.6.4.1	4.4.1		General	See note 1	X
5.6.4.1.1	4.4.1	a		CPR_Decode	E
5.6.4.1.2	4.4.1	b		CPR_Decode	E
5.6.4.2	4.4.2		Input parameters	See note 1	X
5.6.4.2.1	4.4.2	a		See note 1a	X
5.6.4.3	4.4.3		Calculations	See note 1	X
5.6.4.3.1	4.4.3.1	a		See note 1a	X
5.6.4.3.2	4.4.3.2	a		CPR_Decode	E
5.6.4.3.3	4.4.3.3	a		CPR_Decode	E
5.6.5	4.5		Fixed Data Field Position Global Decoding	See note 1	X
5.6.5.1	4.5.1		General	See note 1	X
5.6.5.1.1	4.5.1	a		CPR_Decode	E
5.6.5.1.2	4.5.1	b		CPR_Decode	E
5.6.5.2	4.5.2		Input parameters	See note 1	X
5.6.5.2.1	4.5.2	a		See note 1a	X
5.6.5.3	4.5.3		Transition level straddling	See note 1	X
5.6.5.3.1	4.5.3	a		CPR_Decode	E
5.6.5.4	4.5.4		Calculations	See note 1	X
5.6.5.4.1	4.5.4.1	a		CPR_Decode	E
5.6.5.4.2	4.5.4.2	a		CPR_Decode	E
5.6.6	4.10		Position Report Processing	See note 1	X
5.6.6.1	4.10.1		Services	See note 1	X
5.6.6.1.1	4.10.1	a		See note 2	O
5.6.6.2	4.10.2		Position report parameters	See note 1	X
5.6.6.2.1	4.10.2	a		See note 1a	X
5.6.6.2.2	4.10.2.1	a		See note 1a	X
5.6.6.2.3	4.10.2.1	b		See note 1a	X
5.6.6.2.4	4.10.2.1	c		See note 1a	X
5.6.6.2.5	4.10.2.2	a		See note 1a	X
5.6.6.2.6	4.10.2.2	b		See note 1a	X
5.6.6.3	4.10.3		Position report processing procedures	See note 1	X
5.6.6.3.1	4.10.3.1	a		CPR_Decode	E
5.6.6.3.2	4.10.3.1	b		CPR_Decode	E
5.6.6.3.3	4.10.3.1	c		CPR_Decode	E
5.6.6.3.4	4.10.3.1	d		CPR_Decode	E
5.6.6.3.5	4.10.3.1	e		CPR_Decode	E
5.6.6.3.6	4.10.3.2	a		CPR_Decode	E
5.6.6.3.7				CPR_Decode	E
5.6.6.3.8	4.10.3.3	a		CPR_Decode	E
5.6.6.3.9	4.10.3.3	b		CPR_Decode	E

Annex B (informative): Description of ISO/IEC 9646 test methodology

B.1 Overview of the structure of the ISO/IEC 9646 test-suites

A test-suite covers all tests required to test a piece of equipment. In the ISO/IEC 9646 [7] sense it should consist of the following elements:

- Test-suite overview:

  The test-suite overview presents the general structure of the test-suite. This part primarily contains an index in which the reference between the requirements and the related test cases is outlined.

- Declarations part:

  The declarations part outlines the test environment. Here the test equipment is defined. It also introduces the Points of Control and Observation (PCOs). These points are defined in the test setup where stimuli are injected and were the test results are observed.

- Constraints part:

  The constraints part contains the definitions of the packets and parameters which are used in the test steps. The individual fields of the packets are defined there.

- Detailed test cases (dynamic part):

  The detailed test cases part provides the actual test cases. Each test case is designed for the verification of a distinct function of the test object. In order to allow the performance of individual test cases in any sequence, the test cases are designed to be independent from the history of the test campaign (i.e. they contain all necessary steps required to reach the test objective). Each test case therefore starts at a well defined idle state of the test object. In order to avoid effects on successive test cases each test case must leave the test object in the defined idle state after the execution of the test case.

  A test case consists of a sequence of test steps. Some steps in the beginning of the test case are required to prepare the test object for the actual verification. These steps form the preamble of the test case. The successive steps which perform the actual verification belong to the test body. The steps which bring the equipment under test back to the defined idle state make up the postamble.

B.2 The test case description

ISO/IEC 9646 [7] provides a formal syntax to describe test-suites for communication equipment. This syntax is called the Tree and Tabular Combined Notation (TTCN). The use of TTCN is recommended by ISO/IEC 9646 [7] but not mandated. TTCN is a powerful semi-formal language defined to facilitate computerized test tools for any kind of communication equipment. However, TTCN is, due to its abstractness, not so human friendly as plain text. In order to keep the test cases readable to a maximum extent while making them as formal as necessary, it has been decided to use a simpler formal notation in the description of the test cases.

A more comprehensive description of the syntax follows on the next pages. It is important for the understanding of the test cases to be familiar with the syntax. The following table defines the meaning of entries in individual test cases.

Meaning of entries in the test case table:

Table B.1: Test case format

Field name	**Description**	
Test case name	the name of the test case. This name is used to reference a specific test case in the test-suite.	
Long designator	the long designator directly following the test case name provides the test case scope.	
Purpose	describes the intention of the test case.	
Reference	provides the reference to the clauses of the requirements which are addressed by the tests.	
Context	indicates which part of the test case is executed. The following entries are foreseen:	
	preamble:	in this part of the test case the equipment under test is brought into an appropriate state to begin the actual verification.
	test body:	in this part of the test case the actual test steps required for the verification objective are executed.
	postamble:	in this part of the test case the equipment under test is brought into the defined idle state.
Step	numbers the individual test steps.	
Action	holds the action to be performed during the test.	
	send:	send the specified entity.
	queue:	maintain a queue for input at the specified PCO, respecting any local flow control procedures, so that at least one of the specified entity is always available.
	verify:	verify that a result matches a given outcome (if an outcome is not observed, then the test has failed and the test case must be abandoned !).
	record:	record a value.
	await:	wait until a certain event takes place (the test step has failed if more than 30 s expire before the event is observed !).
	wait:	wait a specified time.
	macro:	execute a named macro.
	do:	do something special which is described in the Action Qualifier column.
	rep*x:* endrep:	repeat the following steps x times in a loop. indicates the end of the loop statements.
	rep*x:* until:	repeat the following steps in a loop until a condition is true. indicates the end of the loop statements and holds the termination condition.
PCO	Point of Control and Observation, which indicates where in the test setup the action shall be performed. The following entries are used:	
	RF	RF antenna connection
	Timing	Timing source input
	Position	Position source input
	Altitude	Altitude source input
	VSS	VSS user
	App in	Application data input
	App out	Application data output
	Self test	Self test passed indication

Field name	Description	
Action qualifier	further qualifies the action. It either holds one or more of the entries shown below:	
		the transaction type to be used together with specific field values. Principally the field values are those presented in the constraints clause. Different field values are stated explicitly like (LCI:= 316 or UD:= [5]{1...5}). The content of data fields which normally consist of several bytes is written like: [n]{val} (e.g. [20]{85}): n bytes with byte value val (decimal values only) [n]{n1...n2} (e.g. [128]{0...127}): n bytes in ascending order from n1 to n2 (decimal values only) [n]{k1,k2,k3,k4,...,kn} (e.g. [5]{4,6,8,10,12}): n bytes according to explicit list (decimal values only)
		the name of a macro plus one or more parameter values required by the macro like: M-NAME (LCI:= 316,CH:= 15).
		a time to wait.
		none, timeout = x s no event to be expected, do not wait longer than x s.
		an event to await.
		parameters of a rep construct in the row with action rep*x* or endrep.
		any free text which further qualifies the action.
	if alternative events are expected in one test step, then they are presented in individual lines but in one row of the table (i.e. only one step number is allocated). Two different cases need to be distinguished: 1) Several events stated in one row without an additional keyword must all appear. Any sequence of the results is valid. 2) Several events combined with an OR may appear alternatively either one or more. Any sequence of the results is valid.	
Ref	A reference to the definition of a basic version of a packet as described in the constraints clause.	
Comment	a comment which adds information for understanding of the actual step.	
Comments	Overall Comments on the test case, if necessary.	

B.3 The queue action

The action "queue" is applied to the VSS User PCO to maintain a constant stream of random access requests. Each request represents a discrete request and results in a single burst with a transmitter ramp up and down at the start and end of the burst. It is not expected that the item under test should be capable of buffering all the random access transmissions demanded by this procedure. The test set should provide a suitable mechanism (e.g. buffer) to maintain a stream of inputs through the VSS User PCO, subject only to the flow control imposed by the item under test.

B.4 The repeat construct

To express test steps which need to be executed repetitively in a loop, the repeat construct is used. A repeat construct consists of the two delimiting keywords:

- **repx**; and
- **endrep**.

In this the parameter "x" stands for the number of loops to be performed. "x" may either be an integer constant or an integer expression. In order to provide the test steps of the loop with possibly required variables, an arbitrary number of variables may be initialized in the Action Qualifier column in the row of the **rep*x*** keyword like:

```
n:= 1;

p( ): =
        {
           CALL REQUEST;
           RECEIVE NOT READY;
           CLEAR REQUEST
        }
```

In the above statements n is initialized to 1. In the second line a vector p( ), holding packets to be used during the loop, is initialized. Each element of the vector may be addressed by an integer index. The first element is addressed by the index 1.

A **rep** statement is used to prepare for a loop of successive statements. There is no test step executed in the rep statement line itself. The loop defined by rep and endrep actually begins in the line following the rep statement line (i.e. the initialization in the rep statement line is only performed once !).

In most loops certain variables need to be modified while the loop is performed several times. The modification is stated by one ore more equation(s) in the **endrep** line, like:

```
n:= n + 1; i:= i -1
```

Nested loops are allowed.

An alternative to the repeat construct which ends after a certain number of loops have been performed is the **repeat until** construct, which consists of the two delimiting keywords:

- **rep*x***; and
- **until.**

In the line with the keyword until the condition is mentioned which terminates the loop. This condition is enclosed by brackets ( ).The parameter x may still be used to indicate a maximum number of loops to be performed. This allows to terminate possible endless loops if the termination condition is not reached due to an error. In such a case the test has failed and must be abandoned!

B.5 Macro definitions

Macros are used to express sequences of steps which are used frequently. A macro may not include verification statements. A macro name is preceded by "M_" for distinction from normal test cases. Macros may be called with parameters. The parameters are mentioned in ( ) behind the macro name the macro is called.

B.6 Test case naming

The individual test cases are named for reference. In order to obtain a systematic name, the name is composed in a hierarchical manner, with subsidiary naming levels separated by the underscore character.

Annex C (informative): Bibliography

EUROCAE ED-108A: "MOPS for VDL Mode 4 Aircraft Transceiver for ADS-B".

List of tables

History

Document history			
V1.1.1	July 2005	Publication	
V1.2.1	July 2006	One-step Approval Procedure	OAP 20061124: 2006-07-26 to 2006-11-24