Technical Note

ANTARES
Communication Standard Technical Specifications

(DRL No: D018B)

Ref: IRIS-AN-CP-TNO-612-ESA-C1
Issue 1.0
27.09.2013

DOCUMENT APPROVAL FORM

Issue History Prepared by	Reference	Change Log	Date
INDRA	ANTAR-B1-CP-TNO-2006-IND	Issue/Revision 1.1	16/09/2013

Approved by	Dissemination level	Date
Technical Officer Oscar del Rio Herrero	Public	27/09/2013

European Space Agency
Agence spatiale européenne

ANTARES

Communication Standard Technical Specifications

(DRL N°: D018B)

Written by	*Responsibility*
Indra Team	Author
Verified by	
J. Batlle (Indra)	WP2 Technical Manager
Approved by	
J.M.Cebrian (Indra)	WP2 Project Manager

Approval evidence is kept within the documentation management system.

Customer Approvals	
G.Raimondo (TAS-I)	Program Manager

CHANGE RECORDS

ISSUE	DATE	§ CHANGE RECORDS	AUTHOR
1.0	03/05/2013	First public document release	Indra Team
1.1	16/09/2013	Minor clarifications and corrections. Section 1.2: Removed incorrect reference to user & control plane external interfaces. Section 2.1.1: Removed "In some cases (when header compression is disabled), it is equivalent to a NPDU." from PPDU definition, as it is misleading (in the case of OSI, L2.5 always adds a 8208 header, even if no header compression). Section 4.1: Minor change about WAN (GS elements “can be interconnected”, instead of “are interconnected”), according to TAS.I RID GPa-003. Section 5.2 / Figure 5-4: IDRP replaced by CLNP & IDRP to clarify that IDRP runs on top of CLNP. Section 11.10: “defined in the following sections” replaced by "defined in the previous sections". Section 8.6.1.1.2: Rewording changes in the definition of TTx: "refers to" replaced by "is associated with". Section 8.7.1.1 renamed to indicate that the section also contains scheduling requirements. Section 8.7.1.2.2: Added specific considerations of the return link ARQ procedure with respect to 8.6.1.1.2 (forward link ARQ procedure). General revision of typos and language errors. Requirements updated/modified as reported in Appendix C.	Indra Team

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

This document, named Communication Standard Technical Specification, has been issued in the scope of WP2 of the ANTARES project and integrates the work performed by the ANTARES WP2 partners.

The main objective of the ANTARES project is the definition of a new Air Traffic Services (ATS) and Airline Operational Control (AOC) satellite Communications Standard and an associated system implementation that allows seamless operation with terrestrial standards.

The new satellite-based Communication Standard, covered by this document, specifies the physical and link layers of the protocol stack, including the mechanisms for signalling between physical entities using the Communication Standard. The Communication Standard specification addresses the user, control and management planes of the standardised interfaces.

1.1 Scope

The present document reports the Communication Standard Technical Specification for the provision of ATS and AOC services through satellite-based communications. The present Technical Specification specifies the physical, link layer and network adaptation layer for the user, control and management planes.

1.2 Structure of the document

This document is structured as follows:

- Section 1 presents the introduction to the document
- Section 2 compiles the definitions, symbols, acronyms and conventions used in the document
- Section 3 reports the applicable and reference documents
- Section 4 reports the satellite communication system overview, identifying the entities involved in the communication
- Section 5 provides an overview of the user, control and management planes
- Section 6 provides the technical specifications for management plane external interfaces
- Section 7 addresses the technical specification for the frequency band, multiple access and system carrier definition, channels and other general requirements
- Section 8 reports the technical specification for the user plane
- Section 9 reports the technical specification for the control plane
- Section 10 provides the technical specification for the management plane
- Section 11 provides the descriptions of the signalling structures

In addition to these sections, the communication standard technical specification also contains the following appendices:

- Appendix A - removed

- Appendix B provides the parity bit accumulators for IRA LDPC codes
- Appendix C summarizes the changes in requirements with respect to the previous version.

2. DEFINITIONS, SYMBOLS, ABBREVIATIONS AND CONVENTIONS

2.1 Definitions

2.1.1 Communication

- **Communication protocol:** A set of rules defining how network entities interact with each other, including both syntactic and semantic definitions.
- **Communication links:**
  - **Mobile link (or ATM link):** communication link between satellite and aircraft (uplink and downlink)
  - **Feeder link:** fixed communication link between satellite and a GSE (Ground Segment Element) for uplink and downlink.
  - **Forward link:** communication link from the Ground Earth Station to aircraft, where:
    - **Uplink** is the communication link from ground to the satellite
    - **Downlink** is the communication link from satellite to aircraft
  - **Return link**: communication link from aircraft to Ground Earth Station where:
    - **Uplink** is the communication link from aircraft to the satellite
    - **Downlink** is the communication link from satellite to ground
- **Protocol stack**: a specific instance of a layered protocol that defines the communication protocol. The present communication standard supports several protocols in parallel, each one using its own terminology. The ISO-OSI reference protocol stack terminology is used for describing these protocols. In the following the description of the layers tailored for the present communication standard is provided:
  - **Physical layer (L1)**: The physical layer defines the Satellite Communication System waveform, including modulation and coding.
  - **Link layer (L2)**: The link layer defines the media access method (often referred to as MAC – Media Access Control) as well as framing, formatting and error control (often referred to as LLC – Link Layer Control).
  - **Network layer (L3)**: The network layer defines the format of end-to-end data packets, as well as routing of packets within the network. The following network layer protocols are supported: ISO 8208 packets (ATN/OSI) and IP (ATN/IPS).
  - **Transport layer (L4)**: The transport layer defines end-to-end functionalities such as reliable/unreliable data transport, flow and congestion control. The transport layer operates end-to-end, and is implemented only in the end systems. Therefore, it has no direct impact on the Satellite Communication system. However, the mechanisms of transport layer have to be carried, in the form of overhead on network layer packets and additional packets.

    The following transport protocols are current or anticipated within the EATM environment:

- TP4 (ATN-OSI reliable transport protocol)
- TCP (ATN-IPS reliable transport protocol)
- UDP (ATN-IPS unreliable transport protocol)

TP4 and TCP generate significant numbers of transport layer acknowledgements.

– **Session layer (L5)**: Not applicable for this standard. Defined by SESARJU.

– **Presentation layer (L6)**: Not applicable for this standard. Defined by SESARJU.

– **Application layer (L7)**: The application layer defines additional mechanisms used by end user applications. It handles user data units in the form of application messages.

Additionally, definition of an additional layer located between layer 3 and layer 2 is useful:

– **Network adaptation layer (L2.5)**: this layer has been specifically defined for the CS and provides functions that are needed to properly adapt the network layer to the generic (i.e., network layer agnostic) CS link and control layer functions. While they require interpretation / knowledge of the specific network layer of the data unit to be transmitted (interpretation of headers, etc.), they do not cover traditional network layer functions as described for layer 3. It includes functions such as IP header compression and support for mobility related events.

- The following definitions are also applicable to the present document (see also Figure 2-1):

  – **NSDU** (Network Service data Unit) **or Segment**: A NSDU is the data unit posted by the transport layer to L3. It is also named Segment.

  – **NPDU** (Network Protocol Data Unit) **or Packet:** A NPDU is the data unit resulting from adding network layer headers to the NSDU provided by L4. It is the basic data unit handled by the network layer and exchanged among L3 peers. It is also the data unit forwarded to the network adaptation layer.

  – **ALSAP** (Adaptation Layer Access point): The ALSAP is the point where L2.5 provides its services to L3.

  – **PNPDU** (Processed Network Protocol Data Unit): A PNPDU is the resulting data unit after the incoming NPDU data unit is processed by the L2.5 layer of the CS (for example, by applying header compression protocols). From a L2 point of view, it is equivalent to a LSDU.

  – **LSAP** (Link Service Access Point): The LSAP is the point where L2 provides the service to L2.5.

  – **LSDU** (Link Service Data Unit) **or Processed Packet:** A LSDU is the data unit posted by the network adaptation layer (L2.5) to L2 at the LSAP.

  – **LPDU** (Link Protocol Data Unit): A LPDU is the data unit resulting from adding L2 headers to LSDU or LSDU fragments (when fragmented).

  – **PSAP** (Physical Service Access Point): The PSAP is the point where L1 provides the service to L2.

  – **PSDU** (Physical Service Data Unit): A PSDU is the data unit posted by L2 to L1 at the PSAP. It is composed of 1 or more LPDUs.

- **PPDU** (Physical Protocol Data Unit): A PPDU is a PSDU data unit which includes also L1 signalling headers (data descriptor), a CRC and padding (in the event that the PSDU size is smaller than the PPDU payload). One PPDU always contains at most one PSDU.
- **PLFRAME** (Physical layer frame) **or Burst**: This is the data unit resulting from physical layer processes (coding, modulation, etc.) as applied to one PPDU. It is also referred to as burst throughout the document.

Note: These terms have been adapted from the OSI layered model terminology.

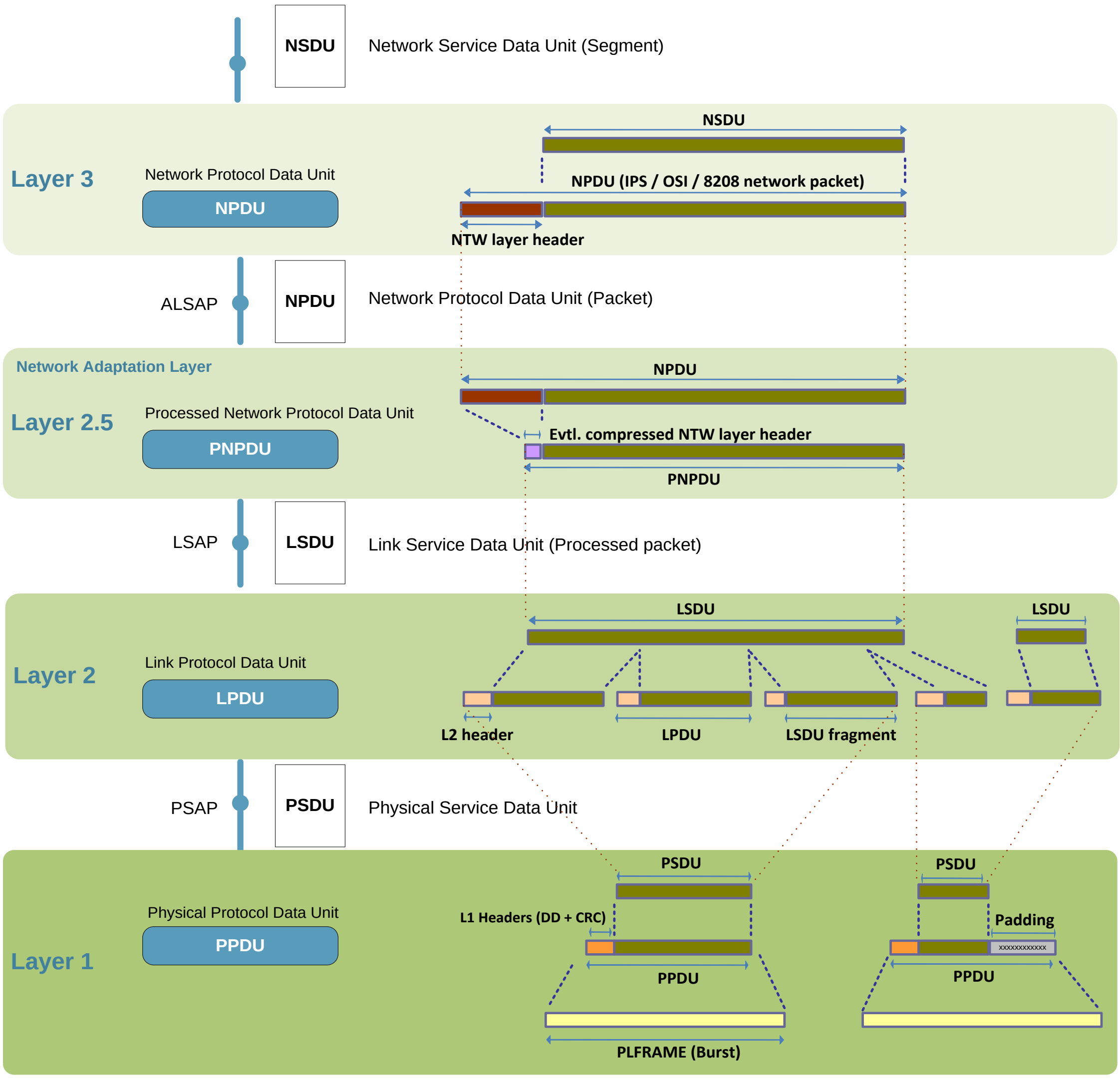


Figure 2-1: Introducing layer model terminology[1]

- **Addressing Terminology**

---

[1] Depicted headers are for illustration only. They refer to headers and trailers and may not always be located at the start of the data unit as shown in the figure. Please refer to the specific data format definitions included in the CS specification for the exact description of the data units.

In order to clarify the description of message contents regarding addresses, here are summarized the different addresses specified at layer 2 and their other names used during the CS specification process:

- **ICAO ID:** 24 bit address fixed per aircraft by ICAO
- **UT ID:** 16 bit address, as defined at ANTARES Layer 2 encapsulation custom scheme, also referred to as UT L2 address
- **GES ID:** 8 bit address, as defined at ANTARES Layer 2 encapsulation custom scheme, also referred to as GES L2 address. The terms GES ID and NCC ID may be used depending on the applicable GSE. The more generic term is GSE ID, referring to either GES ID or NCC ID.
- **GES satellite MAC address:** GES 6 byte address, used for network layer address resolution protocol support in ATN/IPS
- **UT satellite MAC address:** UT 6 byte address, used for network layer address resolution protocol support in ATN/IPS

- **QoS (Quality of Service):**
  - **Flow:** Set of LPDUs belonging to a group of consecutive NPDUs, identified by consecutive sequence numbers/packet counters. The NPDUs' transmission of a flow is performed in order of increasing sequence number/packet counter.
  - **CoS (Class of Service):** Set of applications sharing similar QoS parameters, i.e., continuity (rate of expired NSDUs), ET (expiration time) and TD95 (percentile 95 of transit delay) as defined in SYS-CSY-0080 at [AD-01].
- **Pt-to-pt** (point to point): A point to point channel is transmitted by one source and received by one destination.
- **Pt-to-mp** (point to multipoint): A point to multipoint channel is transmitted by one source and received by several receivers.
- **Mp-to-pt** (multipoint to point): A multipoint to point channel is transmitted by several sources and received by one receiver.
- **Unicast**: The one-to-one transmission of data packets to one specified destination.
- **Multicast**: The one-to-many transmission of data packets to interested destinations.
- **Broadcast**: The one-to-all transmission of data packets to all possible destinations.
- **Physical layer link quality:**
  - **PER (Packet Error Rate):** In the Communication Standard, the term PER refers to the probability that a PPDU is received with errors. Therefore, the PER is computed as the number of erroneous PPDUs divided by the total number of received PPDUs.

2.1.2 Entities

- **Ground Earth Station (GES):** The network entity that provides the feeder link to the Space Segment. In the context of this document, a GES is defined as a logical entity that makes use of communication resources assigned to it in order to communicate with aircraft associated with it, providing voice and data traffic services. A GES will typically use resources within a single beam of a single satellite, but this is not mandatory. A physical site, referred to as Ground Segment Element (GSE), may accommodate several logical GESs, and GESs may share Earth station infrastructure, providing interface with the terrestrial ATM networks.

- **Network Control Centre (NCC):** The network entity that performs the control of the satellite network system resources and elements. It is expected that this entity covers neither the Satellite Control Centre nor the Satellite Operation Centre. In the system reference architecture, one backup NCC located at a different site for satellite link availability purposes for each active NCC can be foreseen.
- **Network Management Centre (NMC):** The network entity that performs the management of the overall satellite communication network (system resources and elements) in a centralised way. It is expected that this entity covers neither the Satellite Control Centre nor the Satellite Operation Centre. Management functions performed by the NMC are considered to be not critical since the system is designed to survive a failure of the management sub-system during a limited time.
- **User Terminal (UT) (also called Airborne Earth Station - AES):** The avionics equipment onboard the aircraft that implements the communication protocol and provides the interface to other on-board elements via an on-board network.

2.1.3 RAMS

- **Continuity**: Probability that a transaction will be completed having met specified performance. Possible anomalous behaviours include late transactions, lost messages or transactions that cannot be recovered within the expiration time, duplicate messages and uncorrected detected message errors.
- **Instantaneous availability:** It is the probability that a service (or system) will be operational (up and running) at any random time, t. This is very similar to the reliability function in that it gives a probability that a system will function at the given time, t. Unlike reliability, the instantaneous availability measure incorporates maintainability information.
- **Average Uptime Availability (or Mean Availability):** The mean availability is the proportion of time during a mission or time period that the system (or service) is available for use. It represents the mean value of the instantaneous availability function over the period (0, T].
- **Integrity:** Integrity is the acceptable rate of transactions that are completed with an undetected error. Undetected errors include undetected corruption of one or more messages within the transaction.
- **Expiration Time:** Maximum time beyond which a NPDU data unit is considered lost.
- **TD95:** 95th percentile of the transit delay one-way latency.
- **Diversity**: The simultaneous use of two or more mutually independent and different systems to increase service availability. Diversity allows improvement in link availability and thus overall system availability by providing (at least) two links thanks to redundant elements and allowing the system to switch from one link to the other one (or combine the two) by choosing the best configuration/link available at a certain time. This solution is mainly used

as a fade countermeasure technique and compensation of the channel, i.e., for events happening outside of the system. The overall availability for the two links (or paths) is better than the one for a single link (or path).

- **Redundancy**: Duplication of one element (of GES, NCC, satellite or UT) or equipment within an element (e.g., RF sub-system) or sub-equipment (e.g., modem) to provide back-up in case of failure. Typical cases to be considered are intended unavailability of equipment due to maintenance operations as well as failures due to design or lifetime of equipment. The objective is to compensate for unavailability of elements due to the system itself.

2.2 Symbols

Symbol	Definition
$C_{ch,a}$	Channelization code of Auxiliary Channel (RTN_ACH)
$C_{ch,d}$	Channelization code of Data Channel (RTN_DCH)
$C_{ch,SF,k}$	Channelization code identifier
$C_{FWD\_I\_SCR}$	In-phase component Physical Layer Complex Scrambling
$C_{FWD\_Q\_SCR}$	Quadrature component Physical Layer Complex Scrambling
$\boldsymbol{c}_{HAD}$	Hadamard Code Block
$c_{HAD\_0}, c_{HAD\_1}, \ldots, c_{HAD\_n}$	Hadamard Code bits
$\boldsymbol{c}_{\mathrm{ldpc}}$	LDPC code word block
C_p	RTN_PREAMBLE channelization code
C_{Scram}	Complex Scrambling code
d(D)	Backward Recursive Systematic Convolutional Generator polynomial
D_{FWD}	Number of rows of FWD Symbol Interleaver
f_N	Nyquist frequency
G	Hadamard generator matrix
G(D)	Transfer function of the 16-state constituent code for PCCC
G(X)	32-bit CRC generator polynomial
$\boldsymbol{h}$	FWD_DD information block
$h_0, h_1, \ldots, h_{23}$	FWD_DD Header information bits
h(D)	Generator polynomial of the binary pseudo-random sequence for the Base-band scrambling
I	In-phase component
$\boldsymbol{i}$	LDPC code information block
$i_0, i_1, \ldots, i_{Kldpc-1}$	LDPC code information bits
$I_{FWD\_PLFRAME}$	In-phase component of a symbol of FWD_PLFRAME
$I_{FWD\_PLFRAME}$	In-phase component of FWD_PLFRAME
$I_{FWD\_S\_PLFRAME}$	In-phase component of FWD_S_PLFRAME
K_{ldpc}	Uncoded LDPC block Size / Number of bits of FWD_BB_FRAME
$\boldsymbol{l}$	FWD_FWD_BD block
$l_0, l_1, \ldots, l_{15}$	RTN_DD Header bits
$l_0, l_1, \ldots, l_m$	FWD_FWD_BD bits
m(k)	Base-band scrambling input sequence (FWD and RTN)
M(X)	Input stream to be processed by the systematic 32-bit CRC

Symbol	Definition
	encoder
m_scr(k)	Base-band scrambling output sequence (scrambled sequence) (FWD and RTN)
M_{FWD}	Number of columns of FWD Bit Interleaver
M_{RTN}	Number of columns of RTN Bit Interleaver
$n_0(D)$	First Forward Recursive Systematic Convolutional Generator polynomial
$n_1(D)$	Second Forward Recursive Systematic Convolutional Generator polynomial
$n_2(D)$	Third Forward Recursive Systematic Convolutional Generator polynomial
N^{ACK}	Number of Acknowledgement
$N_{FEC\_FWD\_BD}$	Number of bits of FEC_FWD_BD
N_{FWD}	Number of rows of FWD Bit Interleaver
$N_{FWD\_BB\_DFL}$	Number of bytes of FWD_BB_DATAFIELD
N_{FWD_BD}	Number of bits of FWD_BD (FWD Burst Descriptor) header
N_{FWD_CRC}	Number of bits FWD_CRC_32 field
N_{FWD_DD}	Number of bytes of FWD_DD (FWD Data Descriptor) header
$N_{FWD\_I\_4XFECFRAME}$	Number of symbols (complex values) of FWD_I_4XFECFRAME
$N_{FWD\_I\_4XFECFRAME\_F}$	Number of symbols of a fragment FWD_I_4XFECFRAME_F
$N_{FWD\_I\_XFRAME}$	Number of symbols per FWD_I_XFRAME
$N_{FWD\_I\_XFRAME\_F}$	Number of symbols in a fragment FWD_I_XFRAME_F
$N_{FWD\_I\_XFRAME\_FN}$	Number of symbols in the last fragment FWD_I_XFRAME_FN
N_{FWD_PB}	Number of pilot symbols in a Pilot Block (PB)
$N_{FWD\_PLFRAME}$	Number of symbols of FWD_PLFRAME
N_{FWD_PPDU}	Number of bits of FWD_PPDU
$N_{FWD\_PREAMBLE}$	Number of FWD_PREAMBLE symbols
N_{FWD_PSDU}	Number of bytes of the incoming FWD_PSDU
$N_{FWD\_S\_PLFRAME}$	Number of symbols of FWD_S_PLFRAME
$N_{FWD\_S\_PPDU}$	Number of bits of Scrambled PPDU
$N_{FWD\_XFECFRAME}$	Number of symbols (complex values) of FWD_XFECFRAME
N_{ldpc}	Coded LDPC block Size / Number of bits of FWD_FECFRAME
$N_{PLHEADER}$	Number of symbols of PLHEADER
N_{RTN}	Number of rows of FWD Bit Interleaver
N_{RTN_ACH}	Number of symbols of Auxiliary Channel (BPSK mapped)
$N_{RTN\_AUXFRAME}$	Number of bits of the RTN_AUXFRAME
$N_{RTN\_BB\_DFL}$	Number of bytes of RTN_BB_DATAFIELD
$N_{RTN\_BBFRAME}$	Number of bits of RTN_BBFRAME
N_{RTN_CRC}	Number of bits of FWD_CRC_32 field
N_{RTN_DCH}	Number of symbols of Data Channel (BPSK mapped)
N_{RTN_DD}	Number of bytes of RTN_DD (RTN Data Descriptor) header
$N_{RTN\_FECFRAME}$	Number of bits of FECFRAME / Coded TCC block size (including tail bits)
N_{RTN_PB}	Number of Pilot bits transmitted through the Auxiliary

Symbol	Definition
	channel (RTN_ACH)
N_{RTN_PPDU}	Number of bits of RTN_PPDU
$N_{RTN\_PREAMBLE}$	Number of RTN_PREAMBLE symbols
N_{RTN_PSDU}	Number of bytes of the incoming RTN_PSDU
$\boldsymbol{p}$	LDPC code parity block
$p_0, p_1, \ldots, p_{Nldpc-Kldpc-1}$	LDPC code parity bits
P_{FWD}	Number of columns of FWD Symbol Interleaver
$\boldsymbol{PH}$	PLHEADER Block
$PH_0, PH_1, \ldots, PH_k$	PLHEADER symbols
Q	Quadrature component
$Q_{FWD\_PLFRAME}$	Quadrature component of a symbol of FWD_PLFRAME
$Q_{FWD\_PLFRAME}$	Quadrature component of FWD_PLFRAME
$Q_{FWD\_S\_PLFRAME}$	Quadrature component of FWD_S_PLFRAME
r	Code rate
R(X)	32-bit CRC (reminder of the division of the input data stream by the 32-bit CRC generator polynomial)
R_1	First (inner) ring Radius of APSK modulation
R_2	Second ring Radius of APSK modulation
$\boldsymbol{s}$	MODCOD_ID block
s_0, s_1, s_2, s_3	MODCOD_ID bits
T_s	Symbol period
$x(0), x(1), \ldots, x(N_{RTN\_S\_BBFRAME}-1)$	TCC (PCCC) code information bits
$y'_1(0), y'_1(1), \ldots, y'_1(N_{RTN\_S\_BBFRAME}-1)$	TCC (PCCC) parity bits of the second encoder (without tail bits)
$y_1(0), y_1(1), \ldots, y_1(N_{RTN\_S\_BBFRAME}-1)$	TCC (PCCC) parity bits of the first encoder (without tail bits)
α	Roll-off factor
β_2	RTN_ACH Gain Factor
β_p	RTN_PREAMBLE Gain Factor
γ	Constellation radius ratio
η_{MOD}	Number of transmitted bits per constellation symbol
$\mu(k)$	Binary pseudo-random sequence for the Base-band scrambling (FWD and RTN)

Table 2-1: Symbols

Note: counters (e.g., Packet Count and Fragment Count) are initialized to zero unless explicitly indicated otherwise.

2.3 Acronym list

Acronym	Definition
16-APSK	16 Amplitude Phase Shift Keying
3GPP2	3rd Generation Partnership Project 2
8-PSK	8 Phase Shift Keying
A-CDMA	Asynchronous Code Division Multiple Access

Acronym	Definition
ACK	Acknowledgement
ACM	Adaptive Coding and Modulation
AD	Applicable Document
AES	Aeronautical Earth Station
AF	Address Format
AGR or A/G-R	Air Ground Router
AMS(R)S	Aeronautical Mobile Satellite (en Route) Service
AOC	Airline Operational Control
APB	Administrative, Political, Business
APSK	Amplitude and Phase-Shift Keying
ARQ	Automatic Repeat reQuest
AS	Autonomous System
ATC	Air Traffic Control
ATM	Air Traffic Management, Asynchronous Transfer Mode
ATN	Aeronautical Telecommunication Network
ATN/IPS	ATN/ Internet Protocol Suite
ATN/OSI	ATN/ Open Systems Interconnection
ATS	Air Traffic Services
A429	Arinc 429
BB	Base Band
BCCH	Broadcast Control Channel
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
BTCH	Broadcast/Multicast Traffic Channel
CBR	Constant Bit Rate
CC	Congestion Control
CDMA	Code Division Multiple Access
CLNP	Connectionless Network Protocol
CCNG	Central Combined NCC GES
CNG	Combined NCC GES
CoS	Class of Service
CRC	Cyclic Redundancy Checksum
CS	Communication Standard
CSP	Communication Service Provider
DD	Data Descriptor
UTCH	Unicast Traffic Channel
ESP	Encapsulating Security Payload
ET	Expiration Time
FC	Fragment Counter

Acronym	Definition
FCAPS	Fault, Configuration, Accounting, Performance and Security
FCH	Forward Channel
FEC	Forward Error Correction
FID	Flow ID
FIFO	First In First Out
FL	Forward Link
FLC	Forward Link Carrier
FMS	Flight Management System
FW	Forward
FWD	Forward
GEO	Geostationary Orbit
GES	Ground Earth Station
GGR or G/G-R	Ground-Ground Router
GMP	Group Management Protocol
GS	Ground Segment
GSE	Ground Segment Element
GW	Gateway
HO	Handover
HPA	High-Power Amplifier
ICAO	International Civil Aviation Organization
ID	Identifier
IDRP	Inter-Domain Routing Protocol
IF	Interface
IP	Internet Protocol
IPS	Internet Protocol Suite
IRA	Irregular Repeat and Accumulate
ISH	Intermediate System Hello
ITU	International Telecommunication Union
L1	Layer 1 – Physical layer
L2	Layer 2 – Link Layer
L3	Layer 3 – Network Layer
L4	Layer 4 – Transport Layer
LDPC	Low Density Parity Check
LPDU	Link Protocol Data Unit
LREF	Local Reference
LPDU	Link Protocol Data Unit
LSDU	Link Service Data Unit
MAC	Medium Access Control
MEO	Medium Earth Orbit

Acronym	Definition
MF-TDMA	Multi Frequency Time Division Multiple Access
MIB	Management Information Base
MODCOD	MODulation and CODification
MSB	Most Significant Bit
MSP	Mobility Service Provider
NCC	Network Control Centre
NCR	Network Clock Reference
NMC	Network Management Centre
NSAP	Network Service Access Point
NTW	Network
OBP	On-board Processor
OSI	Open Systems Interconnection
OVSF	Orthogonal Variable Spreading Factor
PC	Packet Counter
PCCC	Parallel Concatenated Convolutional Code
PDU	Protocol Data Unit
PER	Packet Error Rate
PHB	Per-Hop-Behaviour
PID	Packet ID
PPDU	Physical Protocol Data Unit
PSAP	Physical Service Access Point
PSDU	Physical Service Data Unit
PSK	Phase-Shift Keying
PT	Payload Type
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RA	Random Access
RACCH	Random Access Control Channel
RACH	Random Access Channel
RAMS	Reliability, Availability, Maintainability and Safety
RD	Reference Document
RF	Radio Frequency
RFU	Reserved for Future Use
RHCP	Right-Hand Circular Polarisation
RL	Return Link
RLC	Return Link Carrier
ROHC	RObust Header Compression
RTN	Return
RTP	Real-Time Transport Protocol

Acronym	Definition
RTT	Round-Trip Time
SCC	Satellite Control Centre
SDU	Service Data Unit
SF	Spreading Factor
SNDCF	Sub-Network Dependent Convergence Functions
SNIR	Signal to Noise plus Interference Ratio
SOC	Satellite Operation Centre
SPS	Space Segment
SRRC	Square Root Raised Cosine
SSA	Spread Spectrum Aloha
SSP	Satellite Service Provider
S-WAN	Satellite Wide Area Network
TBC	To Be Confirmed
TBD	To Be Defined
TCC	Turbo Convolutional Code
TCP	Transmission Control Protocol
TD95	Transit Delay 95th percentile
TDMA	Time Division Multiple Access
T-WAN	Terrestrial Wide Area Network
UCCH	Unicast Control CHannel
UDP	User Datagram Protocol
UT	User Terminal
WAN	Wide Area Network

2.4 Specification numbering

The communication standard specification requirements are identified by a unique sequence of three letter codes followed by a number,

AAAA-BBB-CCC-DDDD

where

- AAAA refers to the Document identifier. All the requirements in this document use the code “D018”.
- BBB refers to the nature of the requirement (system or communication standard). All the requirements in this document use the code “COM”.
- CCC refers to the requirement type or class. The following conventions are used:

FUN	Functional requirements

ITF	Interface requirements

– DDDD stands for the requirement numbering

2.5 Verb tense

In the present document, the following definitions apply:

- “Shall”: The specifications formulated as “shall” are mandatory specifications and must be implemented.
- “May”: The requirements formulated as “may” express a permissible practice or action.
- “Will”: The requirements formulated as “will” denote a provision or an intention in connection with this requirement.

3. APPLICABLE AND REFERENCE DOCUMENTS

3.1 Applicable documents

ID	Document Number	Title	Issue	Date
[AD-01]	Iris-B-OS-RSD-0002-ESA	Iris Phase 2.1 System Requirements Document	2.1	23/10/2012

Table 3-1: Applicable documents

3.2 Reference documents

ID	Document Number	Title	Issue	Date
[RD-01]	RFC 2578	Structure of Management Information Version 2 (SMIv2)		04/1999
[RD-02]	Doc 9896-AN/469	Manual for the ATN using IPS Standards and Protocols 1st edition		
[RD-03]	RFC 5795	The RObust Header Compression (ROHC) Framework		03/2010
[RD-04]	RFC 3095	RObust Header Compression (ROHC): Framework and four profiles : RTP, UDP, ESP and uncompressed		07/2001
[RD-05]	RFC 3843	RObust Header Compression (ROHC): A Compression Profile for IP		06/2004
[RD-06]	ICAO Doc 9705/AN956	ATN Standards and Recommended Practices (SARPs). Manual of Technical Provisions for the ATN		
[RD-07]		ITU Radio Regulations, 2008 Edition		2008
[RD-08]	RFC 4815	RObust Header Compression (ROHC): Corrections and Clarifications to RFC 3095		02/2007
[RD-09]	RFC 4996	RObust Header Compression (ROHC): A Profile for TCP/IP (ROHC-TCP)		07/2007
[RD-10]	ICAO Doc 9776	Manual on VHF Digital Link (VDL) Mode 2	Ed 1	2001
[RD-11]	ISO/IEC 13818-1:2000 (E)	Information technology - Generic coding of moving pictures and associated audio information: Systems	Ed 2	2000-12-01

Table 3-2: Reference documents

4. COMMUNICATION STANDARD OVERVIEW

This document is a proposal oriented to an ICAO worldwide approved standard. The CS is intended to be applicable to all aircraft under IFR (including rotorcrafts), distributed and/or centralized GS infrastructures and GEO, HEO and MEO satellite constellations.

4.1 System reference model

The following figure illustrates the system reference model.

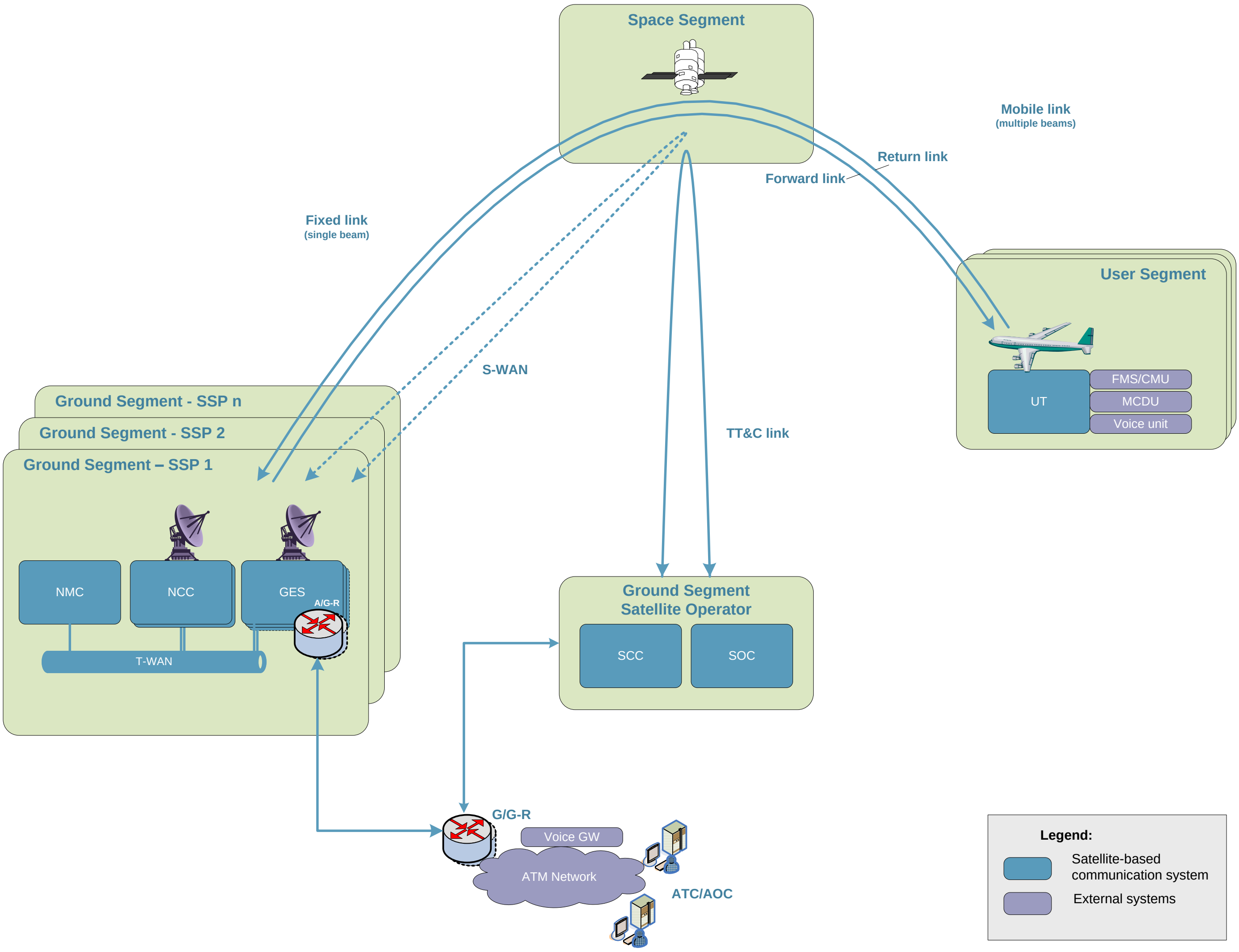


Figure 4-1: System reference model

This figure identifies the physical system components of the satellite communication network as well as the interfaces with the ATM network and the end users (ATC and AOC). The system reference architecture represents the most general system scenario where different Satellite Service Providers (SSPs) participate in the ATM service provision. In this general system configuration each SSP accesses the satellite communication resources by means of multiple stations providing the Tx/Rx communication capabilities. Depending on the particular operational requirements coming from the service providers, a simplified ground segment topology could be envisaged which is characterised by only one SSP accessing the satellite link via a single station. This scenario may be considered a sub-case of the more general one described above.

The satellite communication system reference architecture used for the development of the communication standard is composed of the three following segments:

- **The Space Segment (SPS)**. The SPS is composed of several satellites carrying dedicated ATM payloads operating in the AMS(R)S frequency band (portion of L-band spectrum reserved for aeronautical safety services) in the mobile link, whilst the C, Ku or Ka frequency band is mainly used for the feeder link, Ku band being the most likely one.

  The on-board ATM payloads are transparent, which means that they are not equipped with On-Board Processing (OBP) capabilities. In the mobile link, the SPS is a multi-beam system, while for the feeder link, a single beam is regarded as reference architecture. For the system reference architecture, hot redundant GEO satellites are considered which may be either co-located in the same orbital position or orbitally separated.

- **The Ground Segment (GS)**. The GS is mainly composed of three types of network entities:
  - **Network Control Centre (NCC)**: See definition in section 2.1.2.
  - **Ground Earth Station (GES)**: See definition in section 2.1.2.
  - **Network Management Centre (NMC)**: See definition in section 2.1.2.

  With regard to control functions, it is noted that the communication standard is flexible in order to support SSP GS implementations with different degrees of responsibility distribution between the NCC and GES. In this way, the CS has been designed to be able to cope with a system that has either a centralized or a distributed control architecture.

  In the system reference architecture, multiple GESs are envisaged in order to adapt the satellite communication system to an operation concept where different administrative and political entities (communication service providers, etc.) want to own and operate their facilities with a certain degree of autonomy. In the system reference architecture it is assumed that any GES has connectivity to any mobile beam; that is, a GES can manage UTs located in any mobile beam.

- **The User Segment**. The User Segment is composed of the mobile User Terminals (UTs) (or Aeronautical Earth Stations (AES)); see definition in section 2.1.2. The dimensioning and design of the UT is carried out taking into account several constraints impacting on the final achievable performance. In this respect, constraints on installation (e.g., possible configurations and location of the UT antenna, no need for external active cooling), on regulation, on cost, etc. are considered which influence the final UT performance (e.g., HPA power, antenna radiation pattern, etc.).

Although not considered for CS aspects, the GS will include the **SCC/SOC** stations in charge of controlling and monitoring the satellites.

The CS has been designed to be able to operate without the need for a WAN between the different GSEs, and in consequence providing flexibility to adapt to different GS architectures. However, as shown in the figure, the GS Elements (NMC, NCC and GES) can be interconnected through a Wide Area Network (WAN) supporting the Ground-to-Ground signalling communications.

In this case, the WANs can transport control and management signalling among distant GS elements which may simplify some control procedures.

4.2 Interfaces covered by the Communication Standard

In order to ensure interoperability, the CS defines the air interface of the aircraft from the physical and link layers up to the network layer, both for user and control planes. Additionally, it defines a (logical) interface for management, on the aircraft side. It should be noted that management plane information is not transmitted over the air interface.

Interfaces are shown in Figure 4-2, using dashed lines.

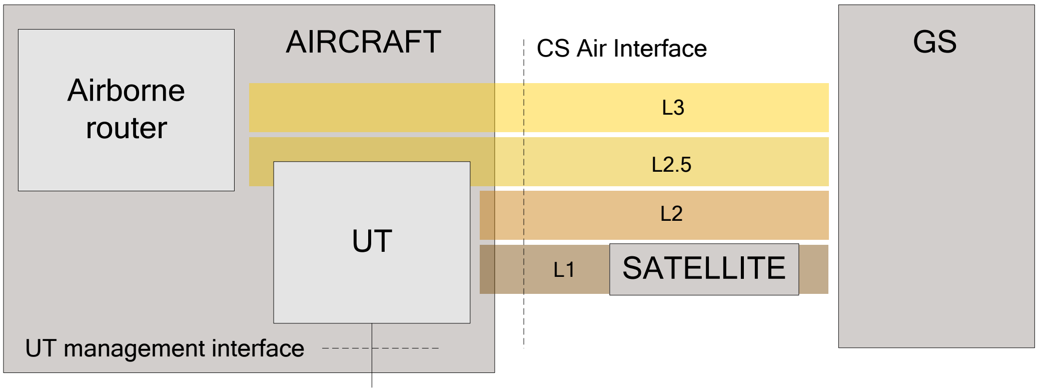


Figure 4-2: Interfaces defined in the CS

The term airborne router refers to the element within the aircraft which supports network layer-related functionality associated with the satellite link interface, whereas the term UT refers to the element which supports at least the CS physical and link layer functions. Layer 2.5 functionality, which provides network layer adaptation functionality, can be provided either by the airborne router, by the UT or by both elements, depending on implementation and also on legacy considerations (in particular, for ATN/OSI support). In any case, the CS has been defined with the intention to not restrict implementation decisions whenever possible as long as this target is compatible with the objective of ensuring proper interoperability.

5. USER, CONTROL AND MANAGEMENT PLANE PROTOCOL STACK OVERVIEW

The following sections illustrate user, control and management plane protocol stacks associated with the CS. It shows a representative distribution of protocol layers in the aircraft and GS elements, but it should be noted that other options could also be fully compliant with the CS.

Elements that are specified fully by the CS are represented in orange, whereas elements which are mainly specified by other standardization bodies are shown in blue.

5.1 User plane protocol stack overview

The following diagram shows a representative user plane protocol stack for ATN/IPS.

In this reference case, the airborne router acts as a mere pass-through (i.e., it is transparent from a network layer point of view), since network layer functionality on the aircraft end is implemented by the FMS, which acts as a mobile node. Additionally, for this particular reference case, layer 2.5 functionality is fully implemented by the UT.

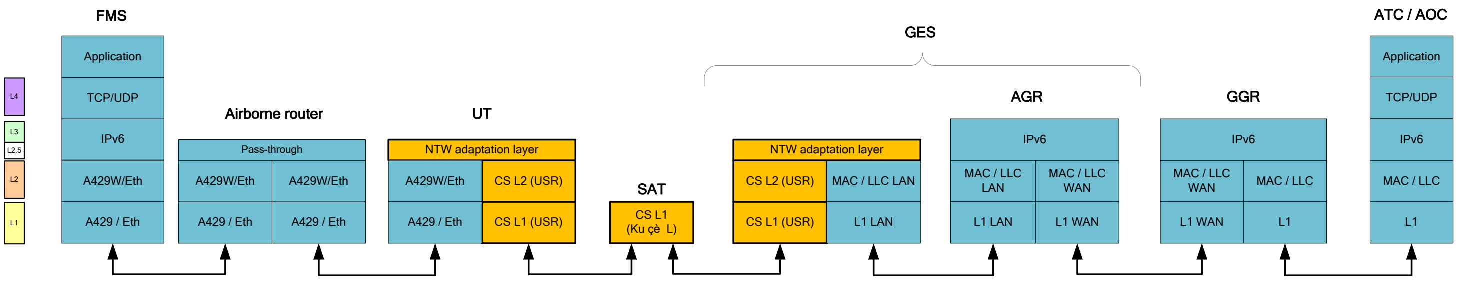


Figure 5-1: Communication standard user plane protocol stack reference (end-to-end) – ATN/IPS

The following diagram shows the expected user plane protocol stack for ATN/OSI.

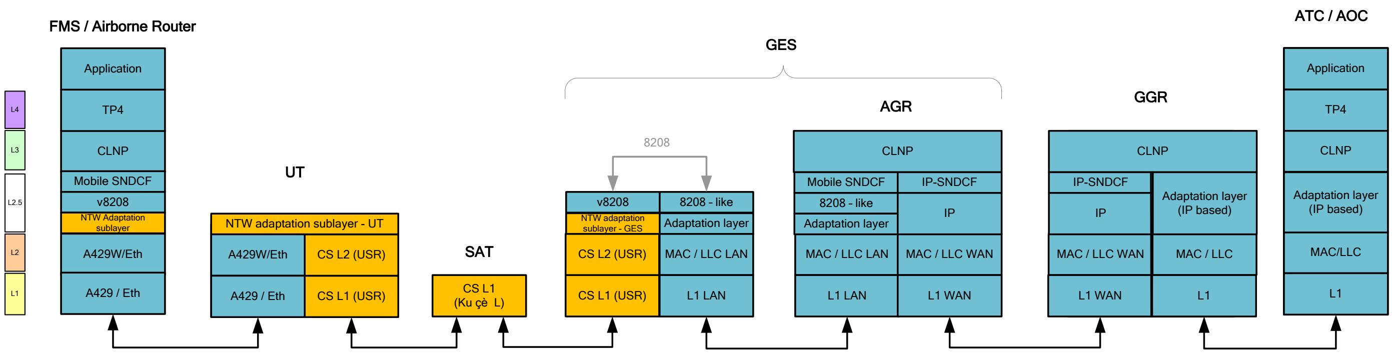


Figure 5-2: Communication standard user plane protocol stack reference (end-to-end) – ATN/OSI

5.2 Control plane protocol stack overview

The following diagrams show the control plane protocol stack for ATN/IPS and ATN/OSI. It includes control functions up to the network layer.

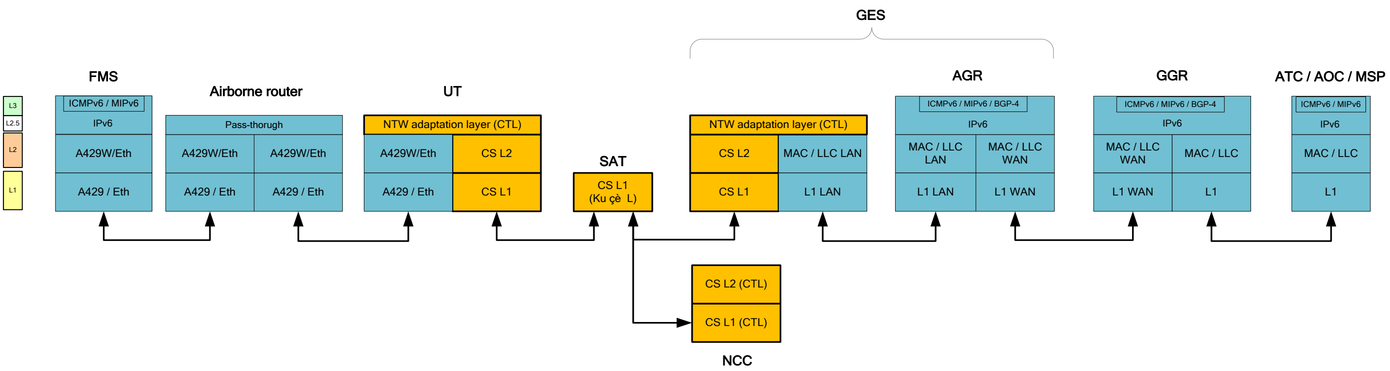


Figure 5-3: Communication standard control plane protocol stack reference – ATN/IPS

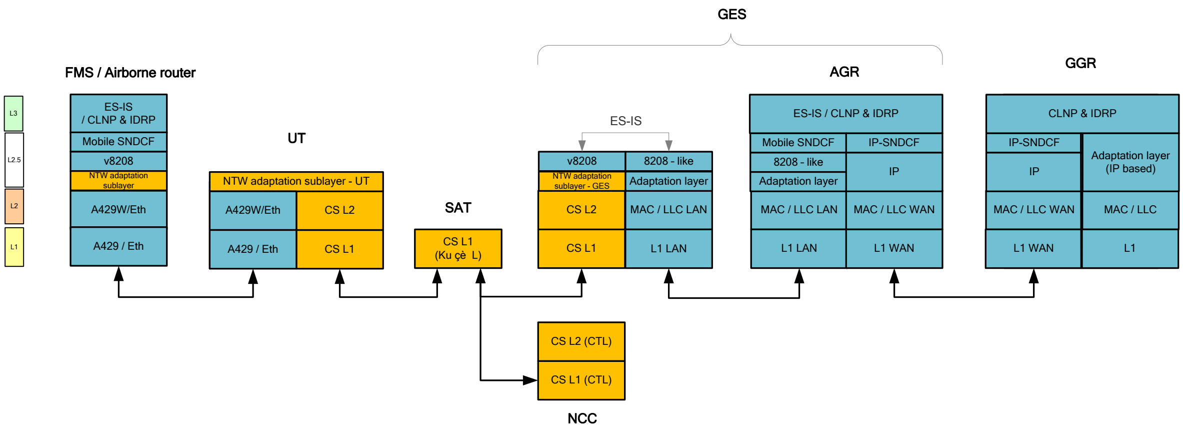


Figure 5-4: Communication standard control plane protocol stack reference – ATN/OSI

5.3 Management plane protocol stack overview

The following diagram shows a possible management plane protocol stack. It should be noted that management plane data is not transmitted over the air interface.

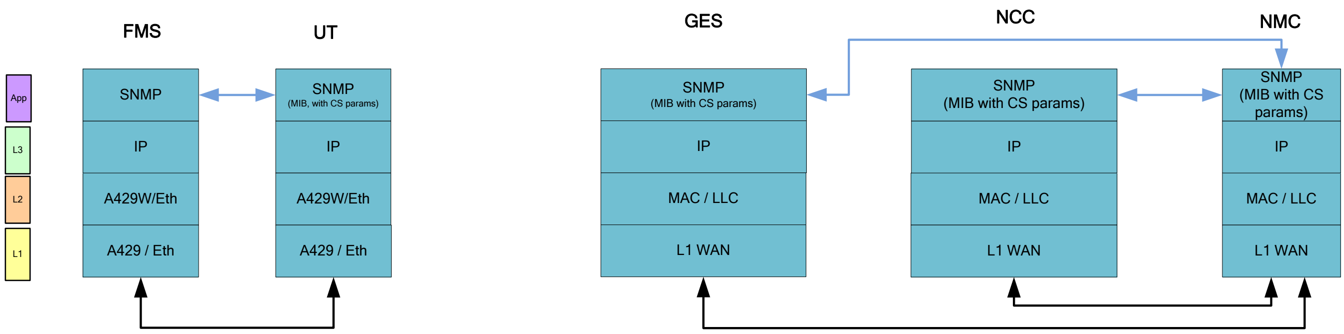


Figure 5-5: Communication standard management plane protocol stack reference

6. MANAGEMENT PLANE EXTERNAL INTERFACES

6.1 General Interface Requirements

D018-COM-ITF-0020

All management element definitions shall be compatible with SMIv2 [RD-01] defined structures.

6.2 UT Management Interface

D018-COM-ITF-0030

At least the following monitored parameters shall be accessible (read-only) from the User Terminal about its UT Satellite Interface:

Parameter	Description	Type
Aircraft ICAO address	Allows access to the aircraft 24-bit ICAO address used in CS procedures.	INTEGER String
L2 addresses	Provides access to the layer 2 address(es) for the User Terminal.	INTEGER String
L3 addresses	Provides access to the layer 3 address(es) configured on the air interface.	INTEGER String
Traffic Log	Indicates the location of the traffic log files on the local file system for retrieval.	INTEGER String
Tx bytes	Provides the number of bytes transmitted on the interface since system startup.	Counter32
Rx bytes	Provides the number of bytes received on the interface since system startup.	Counter32
Rx Power	Indicates the received signal power level (0.1 dBm units).	Integer32
SNIR	Indicates the received Signal-to-(Noise+Interference) Ratio (0.1 dB units).	Integer32
BER	Indicates the current Bit Error rate on the air interface (1E-9 units).	Integer32
PER	Indicates the current Packet Error rate on the air interface (1E-9 units).	Integer32
ModCod evolution	Provides the history of the ModCod used by the user terminal.	INTEGER String
Status	Indicates the status of the communication link. This is a binary status and can be either “up” (1) or “down” (0).	INTEGER
Logon Status	Indicates the logon status of the user terminal. This can either be “success” if the user terminal logon was successful or “error” if not. In the event of error, an error code used to further diagnose the problem will be provided.	INTEGER String
Current GES Address	Indicates the address of the GES to which the User Terminal is currently assigned and to which data packets are to be addressed.	INTEGER String
Current Tx carrier information	Provides access to the details on the current carrier(s) that is in use for transmitting traffic.	INTEGER String

Parameter	Description	Type
Current Rx carrier information	Provides access to the details on the current carrier(s) that is in use for receiving traffic.	INTEGER String
Current SNIR	Indicates the currently published SNIR when ACM is in use (0.1 dB units).	Integer32
Current ARQ retransmissions	Provides the total number of retransmissions performed by the ARQ mechanisms when it is in use.	Counter32
Random Access collision rate	Indicates the current rate of collisions for the random access channel.	Integer32
Current Random Access backoff	Provides access to the current transmission back-off parameter (tx_backoff) for the random access mechanism (10 ms units).	Integer32
Current Random Access persistency	Provides access to the current persistency parameter for the random access mechanism (1E-2 units).	Integer32
Current Random Access timeout	Provides access to the current retransmission timeout parameter for the random access mechanism (10 ms units).	Integer32
Current Data Rate	Provides access to the data rate of the carrier(s) the User Terminal is currently using.	INTEGER String
Fault History	Lists the fault IDs of all the faults that have been recorded since system startup indicating for each fault the timestamp at which it occurred.	INTEGER String
Current Fault Status	Provides access to the current fault status for the User Terminal. Additionally, in the presence of a failed state, information regarding the current fault is also provided.	INTEGER String

Note: for the Type definition, SMIv2 nomenclature has been used.

D018-COM-FUN-2500

A UT shall support the configuration by local management of a set of logon carriers, with at least the following information for each logon carrier (for up to 256 logon carriers):

- Central frequency (entered with a resolution of at least 1 kHz)
- Carrier type (global or regional)

D018-COM-FUN-2510

In the event the UT supports the configuration by local management of system configuration parameters, the following ones shall be supported:

- System ID (4 bits)
- Version number (8 bits)
- RACH burst configuration. List of up to 16 entries, including for each entry the parameters described for the structure RACH_burst_configuration() defined in section 11.

7. GENERAL REQUIREMENTS

7.1 Frequency band, multiple access and system carrier definition

7.1.1 Forward link

7.1.1.1 Mobile link frequency band specifications

D018-COM-FUN-0060

The System shall operate the mobile link at frequencies identified by ITU for Aeronautical Mobile-Satellite (Route) Service (AMS(R)S), in agreement with Article 1, Section III, 1.33 of ITU Radio Regulations [RD-7] and allocated worldwide:

- 1545 to 1555 MHz for the mobile downlink (from satellite to User Terminal)

D018-COM-FUN-0070

The mobile link polarisation shall be right hand circular (RHCP) for uplink and downlink.

7.1.1.2 Forward link multiple access

D018-COM-FUN-0080

The forward link access shall be based on a MF-TDMA scheme.

D018-COM-FUN-0090

The resource segmentation structure of the forward link shall be defined as follows:

- Time-slot: a time duration in which a burst is allocated. It is specified by duration, carrier frequency and carrier rate or bandwidth.
- Frame: defined as a sequence of time-slots at the same carrier.

7.1.1.3 Forward link system carrier definition and burst types

D018-COM-FUN-0100

The Forward Link shall be supported by one type of carrier: Forward Link Carrier (FLC).

D018-COM-FUN-0110

The Forward Link Carrier (FLC) shall provide the frequency and time reference of the system.

D018-COM-FUN-0120

The Forward Link Carrier shall be shared by the GS elements in MF-TDMA mode.

D018-COM-FUN-0130

The Forward Link Carrier shall transmit multiplexed both user traffic and network signalling (broadcast or unicast).

D018-COM-FUN-0135

Multicast user traffic and network signalling shall be transmitted over all GES FLC carriers with potentially interested UTs.

D018-COM-FUN-0137

Broadcast user traffic and network signalling shall be transmitted over all GES FLC carriers.

D018-COM-FUN-0140

The Forward Link Carrier shall transmit multiplexed the following physical channels or burst types (refer to section 7.2.2 for their definition):

- Forward Channel (FCH)

D018-COM-FUN-0150

The Forward Link Carrier shall support the following baud rates:

- 160 kbauds
- 16 kbauds (Low Rate Waveform)

D018-COM-FUN-0160

The two-sided guard band of the forward carriers (FLC) shall be:

- 600 Hz for GEO constellations
- 500 Hz for MEO constellations
- 3.6 kHz for HEO constellations

7.1.2 Return link

7.1.2.1 Mobile link frequency band specifications

D018-COM-FUN-0170

The System shall operate the mobile link at frequencies identified by ITU for Aeronautical Mobile-Satellite (Route) Service (AMS(R)S), in agreement with Article 1, Section III, 1.33 of ITU Radio Regulations [RD-7] and allocated worldwide:

- 1646.5 to 1656.5 MHz for the mobile uplink (from User Terminal to satellite).

7.1.2.2 Return link multiple access

D018-COM-FUN-0180

The return link access shall be based on an A-CDMA (Asynchronous-CDMA) scheme.

7.1.2.3 Return link system carrier definition and burst types

D018-COM-FUN-0190

The Return Link shall be supported by the following type of carrier:

- Return Link Carrier (RLC).

D018-COM-FUN-0200

The Return Link Carrier shall be shared by the UTs in Random Access mode to transmit either signalling or user traffic information.

D018-COM-FUN-0210

The Return Link Carrier shall transmit the following physical channels or burst types (refer to section 7.2.2 for their definition):

- Random Access Channel (RACH).

D018-COM-FUN-0220

The Return Link Carrier shall support the following chip rates:

- 160 kchips/s

D018-COM-FUN-0230

The two-sided guard band of the return carriers shall be 8 kHz.

7.2 Logical and physical channels

7.2.1 Definition of logical channels

The following logical channels, covered in the requirements that follow, are defined below:

	FWD	RTN
Traffic channels		
Broadcast/Multicast Traffic Channel (BTCH)	✓	-
Unicast Traffic Channel (UTCH)	✓	✓
Control channels		
Broadcast Control Channel (BCCH)	✓	-
Unicast Control Channel (UCCH)	✓	✓

Table 7-1: Logical channels

D018-COM-FUN-0240

The BTCH shall be a point-to-multipoint logical channel for the transmission of user plane data to all or a sub-group of UTs within one or more beams.

D018-COM-FUN-0250

The BTCH shall be supported on the Forward Link only.

D018-COM-FUN-0260

The UTCH shall be a point-to-point bi-directional logical channel to be used for the transmission of user plane data (LSDUs) between the GS and a specific UT.

D018-COM-FUN-0270

The BCCH shall be a point-to-multipoint logical channel used to forward general system control information that shall be announced to all or a sub-group of UTs within one or more beams.

D018-COM-FUN-0280

The BCCH shall be supported on the Forward Link only.

D018-COM-FUN-0290

The UCCH shall be a point-to-point bi-directional logical channel used to exchange control messages with a specific UT.

7.2.2 Definition of physical channels

The following physical channels, covered in the requirements that follow, are defined below:

Channel	FWD	RTN
Forward Channel (FCH)	✓	-
Random Access Channel (RACH)	-	✓

Table 7-2: Physical channels

D018-COM-FUN-0300

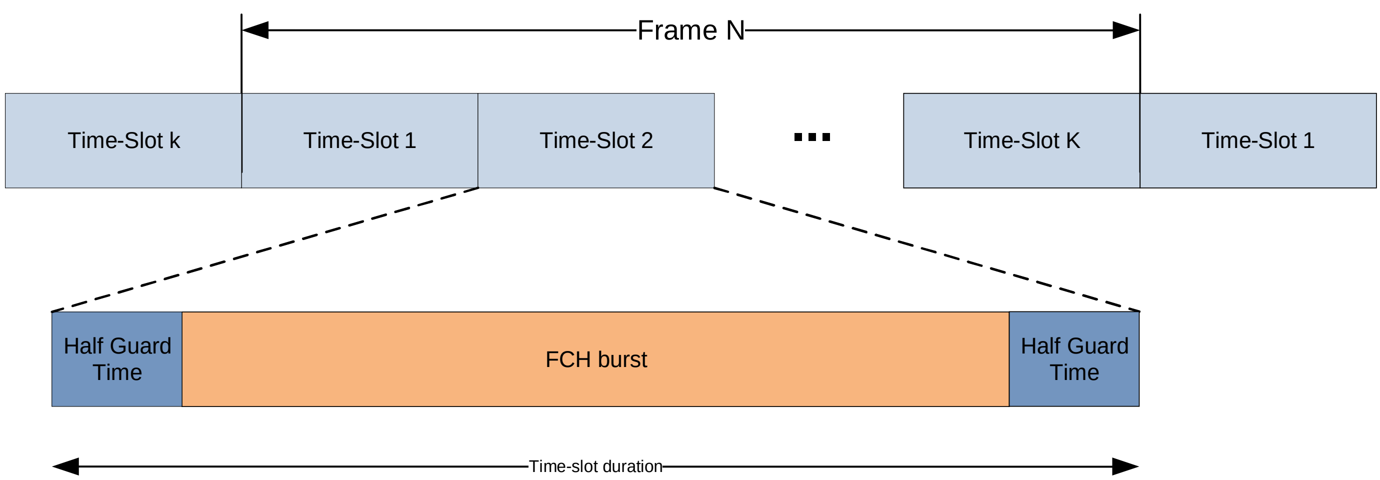
The FCH shall be used for transmitting either control or data information.

D018-COM-FUN-0320

The FCH burst is surrounded by a guard time which allows for power switch-off transients and system timing errors. The guard time of the FCH shall be:

- 20 symbols for FLC at 160 kbaud.
- 13 symbols for FLC at 16 kbaud.

Half the guard time is inserted at the beginning of the time-slot, half the guard time is inserted at the end of the time-slot, as shown in the following figure.



D018-COM-FUN-0330

The RACH shall be used to transfer either control and/or user traffic information.

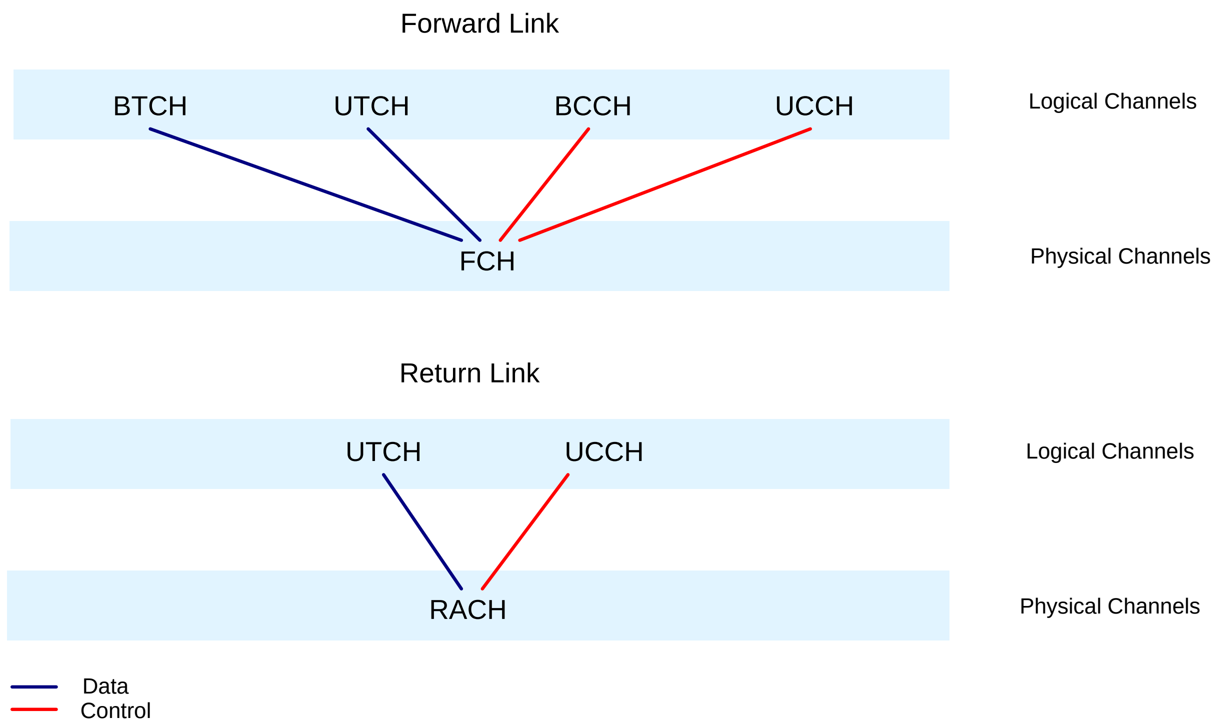
D018-COM-FUN-0340

The RACH shall be a contention-based physical channel whose transmission is based on a Spread Spectrum Aloha (SSA) approach.

7.2.3 Mapping between logical and physical channels

D018-COM-FUN-0360

The mapping between logical channels and physical channels shall be as defined in the following figure:



7.2.4 Link layer process sequencing

D018-COM-FUN-0370

The order of link layer transmission processes shall be as follows:

- Mapping to physical channel(s)
- Queuing / buffering, until the LSDU is scheduled for transmission
- Encapsulation, including associated functions:
  - ARQ header insertion (if applicable)
- MAC procedures associated with random access

D018-COM-FUN-0380

The order of link layer reception processes shall be as follows:

- Link layer filtering
- Reception buffering and re-sequencing, associated with ARQ (if applicable)
- Assembly of LSDU

7.3 Bit order

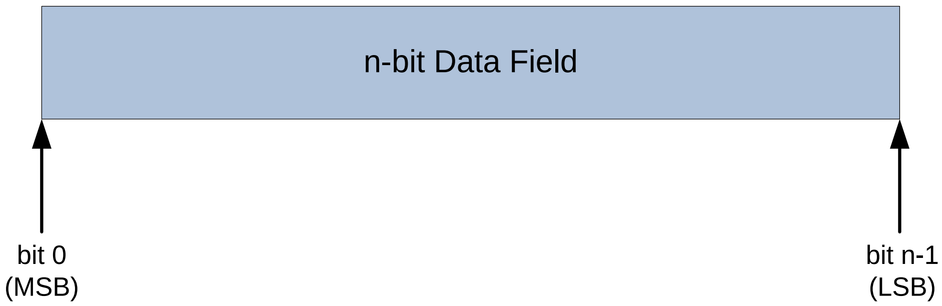
The following specifications are applicable to user, control and management plane.

7.3.1 Bit numbering

D018-COM-FUN-0390

In an n-bit data field, the interpretation of the bit numbering shall be as follows:

- The term "bit n-1" refers to the least significant bit of a multi-bit field
- The most significant bit of a n-bit unsigned field is designated as "bit 0"
- For signed fields, "bit 0" refers to the sign bit and "bit 1" to the most significant bit



7.3.2 Transmission order

The following rules for the transmission order shall be considered in the implementation of the communication standard.

D018-COM-FUN-0400

Unsigned values shall be transmitted starting with the most significant bit and ending with the least significant.

D018-COM-FUN-0410

Signed values shall be transmitted starting with the sign bit, followed by the most significant bit and ending with the least significant bit.

D018-COM-FUN-0420

Bytes shall be processed MSB first (big endian).

8. USER PLANE SPECIFICATION

8.1 User plane description

The User plane covers the aspects related to the transmission of user data. Refer to section 5.1 for details on the user plane.

8.2 Addressing scheme

8.2.1 Unicast and multicast addressing

D018-COM-FUN-0430

The UT shall be identified by a 16-bit L2 address assigned during the logon process. This address should be associated with its 24-bit ICAO address in the ground segment element responsible for the logon procedure while the UT is logged on the system.

D018-COM-FUN-0440

The L2 ranges for UT addressing shall be (in hexadecimal notation):

- from 0x0000 to 0x007F: Multicast addresses
- from 0x0080 to 0x00FF: Reserved addresses for ground to ground communications.
- from 0x0100 to 0xFFFD: Unicast addresses
- 0xFFFE: Dummy address
- 0xFFFF: Broadcast address

UTs shall silently discard the payload of any LPDU using the Dummy address.

D018-COM-FUN-0450

Each GS element shall be assigned a unique 8-bit L2 address used for all communications in which L2 addressing is required.

D018-COM-FUN-0460

The L2 ranges for GS element addressing shall be (in hexadecimal notation):

- from 0x00 to 0x7F: Reserved addresses
- from 0x80 to 0xFE: Unicast addresses
- 0xFF: Reserved address

D018-COM-FUN-0463

Each UT shall be assigned a fixed and unique 6-byte MAC address for the satellite link-layer interface, with up to 4 MAC addresses per UT.

D018-COM-FUN-0468

Each GSE shall be assigned a fixed and unique 6-byte MAC address for the satellite link-layer interface.

D018-COM-FUN-0469

The UT shall identify a GSE by using the first six most significant bits of the GSE ID field, except for the link layer ARQ and encapsulation processes.

Note: The last two least significant bits of the GSE ID field may thus be used to distinguish the active GSE from a number of redundant stations transparently to the UT.

8.2.2 Multicast addressing

Section removed.

8.3 Security

Security authentication procedures specified at SRD requirement COM-GEN-0150 to be supported are basically transparent to the CS. Whenever an authentication procedure at upper layers happens, the communication system will convey the upper layer authentication procedure messages like any other application, making the satellite communication system basically unaware of the event, since it is carried out by application layer entities.

8.4 Network layer support

8.4.1 ATN/IPS

D018-COM-FUN-0500

The CS network adaptation layer shall support the IPv6 network layer protocol as specified by ICAO in [RD-02].

8.4.1.1 Mapping between link layer and network layer addresses

D018-COM-FUN-0520

If an UT provides multiple link-layer interfaces, then an IPv6 packet received by the UT over the satellite link shall be forwarded to the correct interface based on the destination Care of Address or link-local IPv6 address.

Note: This requirement addresses a scenario with the UT providing multiple link-layer interface to multiple mobile nodes. It is feasible as the traffic flow is always air initiated (i.e., the first NPDU having the CoA or LLA as L3 source address is always sent by the aircraft).

D018-COM-FUN-0530

The CS network adaptation layer shall include the source link layer address option in the Router Advertisement messages (as defined in RFC4861), which are sent by the GS.

D018-COM-FUN-0533

The Source/Target Link-layer Address option defined in RFC4861 shall have a Length field value of 1 and the Link-Layer Address shall be configured with the MAC address mentioned in D018-COM-FUN-0463 and D018-COM-FUN-0468, as applicable.

D018-COM-FUN-0536

The Interface Identifier for a CS link-layer interface shall be based on the EUI-64 identifier derived from the MAC address mentioned in D018-COM-FUN-0463. This EUI-64 is formed in the same way as indicated in RFC2464 section 4.

D018-COM-FUN-0538

The IPv6 link-local address assigned to the CS link-layer interface is formed by appending the Interface Identifier, as defined in D018-COM-FUN-0536, to the prefix FE80::/64, in the same way as specified in RFC2464 section 5.

D018-COM-FUN-0540

On the FWD link, an IPv6 packet with a multicast destination address DST, consisting of the sixteen octets DST[1] through DST[16] (last byte), shall be transmitted using a destination UT ID with the last 7 bits equal to the last 7 bits of DST[16] and the remaining bits set to zero.

D018-COM-FUN-0547

On the RTN link, an IPv6 packet with a multicast destination address shall be transmitted using as destination the (unicast) GES ID defined in D018-COM-FUN-0450 (if it is to be included in the encapsulation header).

Note: The GES ID is included in the encapsulation header if indicated by the Include_destination_address_flag of the LOGON VALIDATION ACCEPT or HO VALIDATION signalling messages.

8.4.2 ATN/OSI

D018-COM-FUN-0550

The CS network adaptation layer shall support the OSI network layer as defined in ATN/OSI SARPS [RD-06] for air-ground (mobile) subnetworks.

8.4.2.1 Mapping between link layer and network layer addresses

In ATN/OSI, mapping between 8208 and CLNP network layer addresses is provided through the exchange of ISH protocol data units, as specified in [RD-06].

D018-COM-FUN-0560

In the GS, all transmitted 8208 packets associated with a certain virtual circuit shall be forwarded over the same CS logical channel where the 8208 CALL-REQUEST was received.

D018-COM-FUN-0570

In the aircraft, all transmitted 8208 packets associated with a certain virtual circuit shall be forwarded over the same CS logical channel where the JOIN was received.

8.5 Network layer adaptation functions

8.5.1 ATN/IPS

8.5.1.1 QoS

D018-COM-FUN-0580

For the FWD and RTN, link data and signalling transmissions scheduling the CS network adaptation layer shall contain the functionalities (enabled vs. disabled) to inspect transport layer port numbers, transport layer protocol type and network layer type of service fields (e.g., ToS or DSCP for ATN/IPS) in order to infer the next QoS information:

- Application type
- Application layer message length in bytes
- TD95 (s)
- ET (s)
- Continuity
- Integrity

D018-COM-FUN-0581

In the event that no information in regards to QoS is available, the CS link layer shall use the default Best Effort QoS class for all NPDUs received from the network adaptation layer (lowest priority CoS).

D018-COM-FUN-0590

The UT and GES shall keep a register of the next QoS measurements per application, in order to measure its compliance with required application QoS:

- TD95 of NPDU latencies
- Number of NPDU successfully transmitted vs. arrived (continuity estimation)

8.5.1.2 Compression

8.5.1.2.1 Header Compression

D018-COM-FUN-0600

The UT shall implement RObust Header Compression framework (ROHC) as specified in [RD-03].

D018-COM-FUN-0610

The GS shall implement RObust Header Compression framework (ROHC) as specified in [RD-03].

D018-COM-FUN-0620

The following ROHC profiles are mandatory:

- Uncompressed profile (defined in [RD-03] and [RD-04])
- RTP profile (defined in [RD-04] and [RD-08])

D018-COM-FUN-0625

The following ROHC profiles are optional:

- UDP profile (defined in [RD-04] and [RD-08])
- IP-Only profile (defined in [RD-05] and [RD-08])
- TCP profile (defined in [RD-09])

D018-COM-FUN-0630

The following ROHC configuration parameters shall be set as indicated:

- MAX_CID = 16383
- LARGE_CIDS = 0x01 (True)
- MRRU = 0 (ROHC packet segmentation is disabled)

8.5.2 ATN/OSI

8.5.2.1 SARPs network layer adaptation functions

D018-COM-FUN-0660

The CS network adaptation layer shall support the Mobile SNDCF, as defined in [RD-06].

D018-COM-FUN-0670

The CS network adaptation layer shall support the v8208 SNAcP, as defined in [RD-10].

D018-COM-FUN-0680

Upon successful finalization of the logon procedure, a new v8208 virtual circuit shall be established by the aircraft over the newly established link for each priority class, including as destination SNAP address one of the 8208-AGR-Addresses which are part of the LOGON VALIDATION ACCEPT message.

D018-COM-FUN-0690

Upon reception of a valid HANDOVER VALIDATION message associated with a handover procedure of type 1 and 2, a new v8208 virtual circuit shall be established by the aircraft over the newly established link for each priority class, including as destination SNAP address one of the 8208-AGR-Addresses which are part of the HANDOVER VALIDATION message.

8.5.2.2 QoS

The same requirements as defined in section 8.5.1.1 apply for ATN/OSI traffic. It should be noted that although ATN/OSI supports different priorities, currently deployed ATN/OSI based services all use the same priority.

8.5.2.3 Compression

8.5.2.3.1 Header compression

D018-COM-FUN-0710

LREF header compression shall be implemented as specified in the ICAO ATN SARPS [RD-06].

8.6 User plane forward link specification

8.6.1 Link layer specification

D018-COM-FUN-0715

For the FWD link ARQ and encapsulation processes, all the bits of the GSE ID field shall be used.

Note: This requirement has been included for redundancy support purposes. Refer also to D018-COM-FUN-0469 and D018-COM-FUN-1601.

D018-COM-FUN-0716

For a certain link, the UT shall be able to support the FWD link ARQ and reassembly process at least for four different GSE ID values.

8.6.1.1 ARQ protocol

8.6.1.1.1 ARQ header format

D018-COM-FUN-0720

The ARQ header format shall include the following fields:

Flow ID	Packet Count	Fragment Count	P/F
4 bits	4 bits	0/6 bits	0/1 bits

where

Field	Length	Description
Flow identifier field	4 bits	This field differentiates between different flows.
Packet Count field	4 bits	This field is a counter of packets associated with a flow. It is incremented for each packet sent toward a specific destination, using an independent counter for each flow.
Fragment Count field	0/6 bits	This field identifies individual fragments (LPDUs) of a NPDU. This field is only required in LPDUs of fragmented NPDUs.
P/F bit field	0/1 bit	This field indicates whether the LPDU shall be acknowledged by the receiver ('1') or not ('0'). This bit shall be set to '0' for NPDUs not requiring ARQ support. This field is not present in un-fragmented NPDUs. The receiver shall acknowledge un-fragmented NPDUs when they require ARQ support.

D018-COM-FUN-0730

The ARQ ACK packet in the RL acknowledging LPDU sent through the FL shall include the following fields:

Flow ID	Packet Count	R	Fragment Count
4 bits	4 bits	2 bits	6 bits

where

Field	Length	Description
Flow identifier field	4 bits	This field is a copy of the Flow Identifier of the acknowledged LPDU.
Packet Count field	4 bits	This field is a copy of the Packet Counter of the acknowledged LPDU.
Fragment Count field	6 bits	This field is a copy of the Fragment Counter of the acknowledged LPDU. In the event that the acknowledged LPDU belongs to an un-fragmented NPDU, this value shall be set to 0.
R field	2 bits	Reserved for future use. Shall be set to 0.

This information, along with the L2 addresses, shall be encapsulated in a RL link layer packet which shall be transmitted without ARQ support as specified in D018-COM-FUN-3773.

8.6.1.1.2 ARQ procedure

This section provides the specification of the forward link ARQ procedure.

The following definitions are used to support the description of the procedure:

- *ARQ Block*: The ARQ block is the concept used to refer to the consecutive LPDUs, belonging to a NPDU, for which the transmitter expects an acknowledgement. ARQ block length in the case of multi-fragment NPDUs is an implementation decision.
- T^{Tx} refers to the instant at which the PPDU payload containing the LPDU that T^{Tx} is associated with is transmitted on the uplink. It is used as the reference for the time interval within which the transmitter expects to receive an ARQ-ACK validating an ARQ block (i.e., *arq_retx_timeout).*
- *arq_retx_timeout* refers to the time interval within which the transmitter expects to receive an ARQ-ACK. The value of *arq_retx_timeout* is system-dependent and is not standardized. The UT receiver is agnostic to this value, so it is not necessary for the GS to distribute this value via system signalling.

The timer construct SET is used to activate a timer. The SET construct takes two parameters. The first parameter is the instant (absolute time) for the expiration of the timer. The second one is the timer identifier. The timer generates a timeout event when the time reaches the previously specified time.

The timer construct RESET is used to deactivate a timer. The RESET construct takes as unique parameter the timer identifier. After resetting the timer, the process behaves as if the timer has never been activated.

D018-COM-FUN-0740

The forward link ARQ procedure shall comply with the rules stated hereafter:

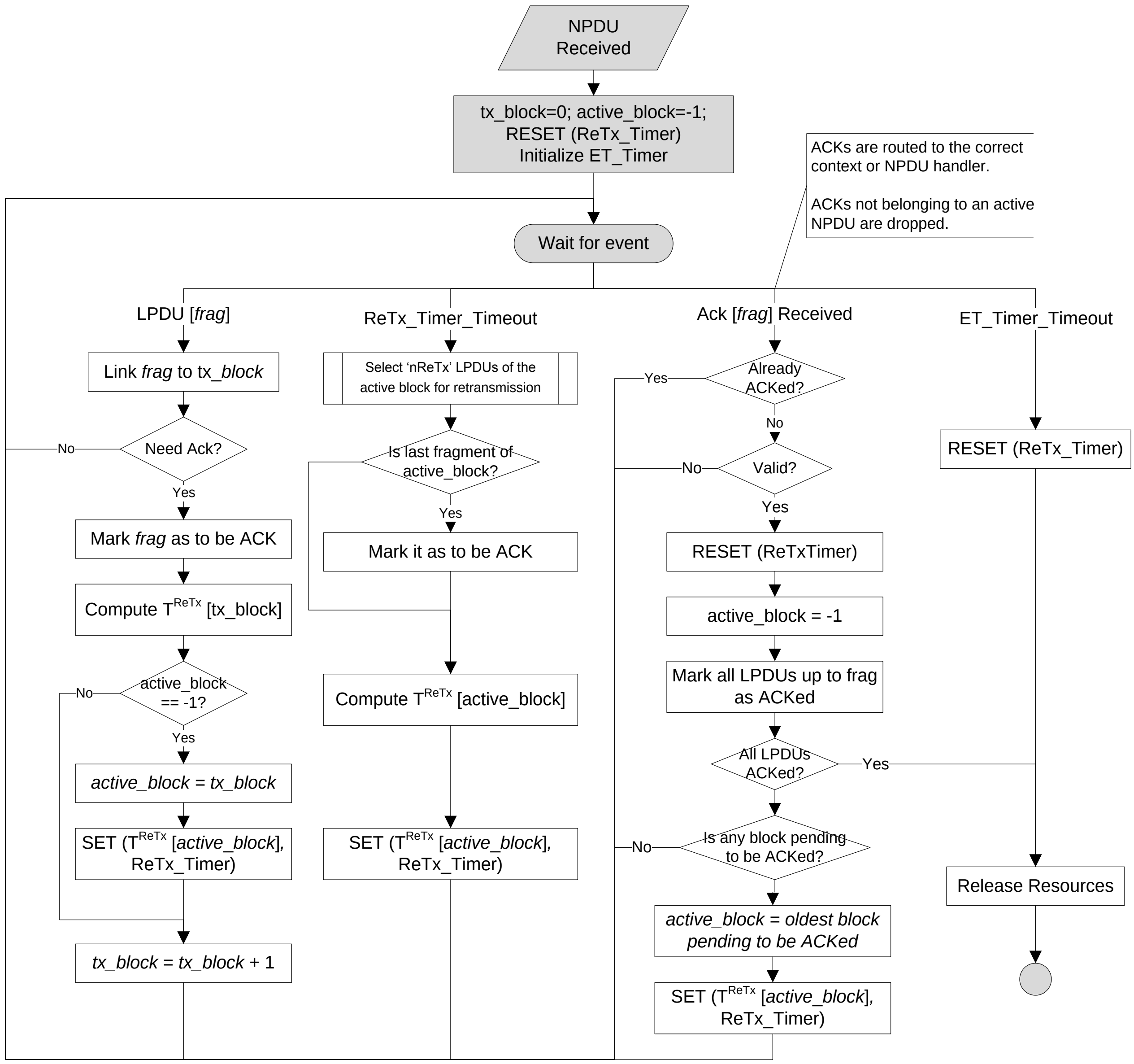
- The ARQ protocol only provides support to unicast traffic
- The ARQ protocol univocally resolves which NPDU an LPDU or ARQ-ACK belongs to using the link layer information below:
  - The UT ID.
  - The identifier resulting after masking the GSE ID with the bit pattern specified in D018-COM-FUN-0715.
  - The Flow Identifier and Packet Counter from the ARQ field in encapsulation header (D018-COM-FUN-0720).
- The P/F bit in the ARQ field of the encapsulation header indicates which LPDUs the receiver is expected to acknowledge. The P/F bit is always set by the transmitter in the last LPDU of multi-fragment NPDUs. The LPDU belonging to a single-fragment NPDU does not contain P/F field in the header, and is always acknowledged by the receiver.
- The ARQ receiver generates a positive cumulative ARQ-ACK upon reception of an LPDU with the P/F bit set.
- The ARQ receiver generates a positive cumulative ARQ-ACK upon reception of a LPDU with S, E and ARQ bits all being set in the encapsulation header.

- The reception of an ARQ-ACK acknowledges correct reception of all transmitted LPDUs with a fragment count equal to, or lower than, the fragment count indicated in the ARQ-ACK.
- Retransmissions are triggered by timeouts at the transmitter end.
- The transmitter prevents the reordering of NPDUs within the same flow.

8.6.1.1.2.1 ARQ procedure at transmitter end

D018-COM-FUN-0750

The forward link ARQ procedure at the transmitter end shall comply with the SDL diagram and the verbal walkthrough shown below.



The process starts with the reception of a NPDU requiring ARQ support for its transmission. A new context is created to handle its transmission. The following support variables are initialized and tagged to the NPDU at this stage:

- ***tx_block***: This variable tracks the ARQ block that is being transmitted. This variable is initialized to 0.
- ***active_block***: This variable tracks the ARQ block for which the transmitter is handling the retransmission at a given time. This variable is initialized to -1.
- T^{ReTx} *[]*:This array tracks, for each ARQ block, the instant before which the transmitter expects to receive an ARQ-ACK validating the ARQ block that T^{ReTx} refers to. This variable is initialized to 0.
- ***ReTx_Timer***: This timer is used to track the latest instant that the transmitter waits for an ARQ-ACK validating the correct reception of the LPDUs of the *active_block* before triggering its retransmission. Only one *RxTx_Timer* instance per NPDU can be active simultaneously. It is RESET during the initialization.
- ***ET_Timer***: This timer is used to track the latest instant before which the transmission of an NPDU must complete before violating the ET QoS constraint. Upon expiration, the NPDU is de-queued and its resources freed.
- ***nReTx*** refers to the number of LPDUs out of the *active_block* that the transmitter selects for retransmission upon expiration of *ReTx_Timer*. *The ReTx_Timer* is SET after the last LPDU of the selected set of LPDUs is transmitted. *Typica*lly, all the LPDUs of the *active_block* are *selected f*or retransmission at once. Nonetheless, the transmitter can select subsets of LPDUs while it guaranties that all LPDUs in the *active_block* have a chance of retransmission until the ARQ-ACK is rec*eived.*

Operational Phase

During the operational phase, the ARQ process is waiting and responding to the events below:

- ***LPDU [frag]*:** This event indicates that the LPDU with fragment counter '*frag'* is created:
  - The LPDU is associated with the current *tx_block*.
  - The transmitter decides whether or not to request acknowledgement for the LPDU:
    - It is mandatory to request acknowledgement for the last LPDU of an NPDU.
    - For other LPDUs, it is an implementation decision.
  - If the LPDU needs to be acknowledged:
    - The LPDU becomes the last LPDU of *tx_block*.
    - The time of the retransmission of *tx_block* is computed and tagged to *tx_block* taking as a reference T^{Tx}, the time at which the LPDU is actually transmitted on the uplink.

$$T^{ReTx}[tx\_block] = T^{Tx} + arq\_retx\_timeout$$

- In the event that there are no ARQ blocks waiting to be acknowledged, *active_block == -1:*
  - The *active_block* variable is set to *tx_block.*
  - The *ReTx_Timer* is SET to T^{ReTx} *[active_block]*.
- The variable *tx_block* is incremented by one.

– ***<u>ReTx_Timer_Timeout</u>***: This event indicates that *ReTx_Timer* expired because an ARQ-ACK validating *active_block* has not been received before the expected time.

- The transmitter starts the retransmission of the LPDUs associated with the *active_block* by selecting '*nReTx*' LPDUs out of the LPDUs belonging to the *active block*.
  - The LPDUs retransmitted re-use the ARQ field used for its initial transmission.
  - The time of the retransmission is computed and tagged to *active_block* taking as a reference, T^{Tx}, the time at which the last of the selected LPDUs is actually transmitted on the uplink.

    T^{ReTx} *[active_block]* = T^{Tx} + *arq_retx_timeout*
  - The *ReTx_Timer* is SET to T^{ReTx} *[active_block].*

– ***<u>Ack [Frag] Received</u>***: This event indicates the reception of an ACK for the LPDU with fragment counter '*frag*'. *<u>ACKs are forwarded to the suitable context, or dropped in the event that no active context is found</u>*.

- The received ARQ-ACK is silently discarded in the event that it belongs to an LPDU already acknowledged. Otherwise,
- The received ACK is discarded if:
  - It refers to an LPDU which is not marked to be acknowledged.
  - It refers to a not yet transmitted LPDU.

Otherwise, the received ARQ-ACK acknowledges some still un-acknowledged LPDUs, so:

- The variable *active_block* is set to '-1' and the *ReTx_Timer* is RESET.
- When all the LPDUs of the NPDU are acknowledged, the NPDU is regarded as successfully received and their resources are released.

Otherwise, there are still LPDUs pending to be acknowledged, in which case;

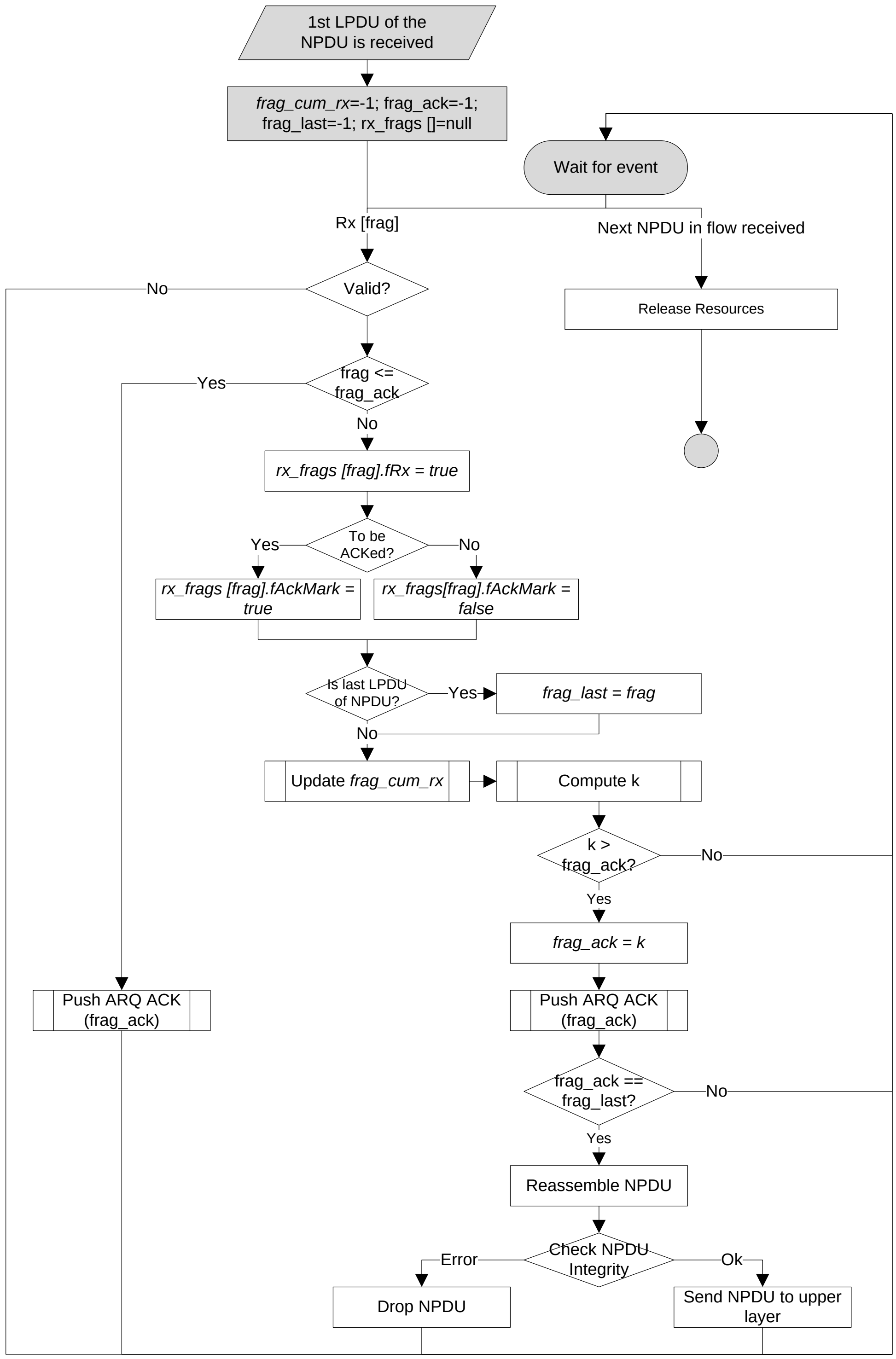
- The process seeks the oldest fully transmitted ARQ block which is not yet acknowledged.
- If such an ARQ block is found, *active_block* is set to such a block and the *ReTx_Timer* is SET to *TReTx [active_block]*. If the instant T^{ReTx} [*active_block*] is already in the past, the *ReTx_Timer* triggers immediately.

- ***ET_Timer_Timeout***: This event indicates that the time allowed for the NPDU transmission has expired. Hence, its transmission is aborted, *ReTx_Timer* is RESET and the allocated resources are released.

8.6.1.1.2.2 ARQ procedure at receiver end

D018-COM-FUN-0760

The forward link ARQ procedure at the receiver end shall comply with the SDL diagram and the verbal walkthrough shown below.



The process starts with the reception of a LPDU that does not belong to an already allocated NPDU. In response, a new context is created and the following support variables are initialized:

- ***rx_frags []***: This is an array containing information about the LPDUs received. Each entry of the array contains the following information:
  - fRx: A flag indicating whether the LPDU has been received.

- fAckMark: A flag indicating whether the receiver is requested to generate acknowledgement for the LPDU. The receiver is requested to generate acknowledgement for:
  - Those LPDUs with S, E and ARQ bits that are all set in the encapsulation header, and
  - Those LPDUs that have the P/F bit in the ARQ field set by the transmitter.

– ***frag_cum_rx***: This variable tracks the fragment counter of the largest consecutive LPDU already received. This variable is initialized to -1.

– ***frag_ack***: This variable tracks the fragment counter of the ACKed (the ARQ-ACK has been sent) LPDU with the greatest fragment counter. This variable is initialized to -1.

– ***frag_last***: This variable tracks the fragment counter of the last LPDU of the NPDU. This variable is initialized to -1. *The receiver identifies the last LPDU of a NPDU using the 'E' bit of the L2 header*.

Within the scope of this definition, the term "acknowledged" is used to refer to the generation of ARQ-ACKs at the receiver end.

Operational Phase

During the operational phase, the ARQ process is waiting for the events below:

– ***Rx [frag]***: This event indicates the reception of an LPDU with fragment count '*frag*'.

- The LPDU is dropped in the event that the validity check fails because:
  - There is any inconsistency in the L2 header. *This process should be performed in earlier phases although it has been included in the diagram and description for the sake of completeness.*

  Otherwise, it is checked if the just received LPDU is regarded as already acknowledged, *frag ≤ frag_ack*:

  - If so, a new ARQ-ACK is generated using the value *frag_ack* as the fragment counter field.

  Otherwise,

- The *rx_frags [frag]* entry is updated as indicated below:
  - *rx_frags [frag].fRx = true*
  - *rx_frags [frag].fAckMark* is set to *true* in the event that the receiver is requested to generate acknowledgement for the LPDU or to 'false' otherwise.
- The variable *frag_last* is set to *frag* in event that the received fragment belongs to the last fragment of the NPDU.
- The value of the variable *frag_cum_rx* is updated as indicated below:

while (rx_frags [frag_cum_rx+1].fRx == true) frag_cum_rx += 1

- The procedure computes 'k' by seeking within the updated set of consecutive LPDUs already received, i.e., *frag_cum_rx*, for the LPDU with the greatest fragment counter that is requested to generate an ARQ-ACK.

  for (k=frag_cum_rx; (k>frag_ack) && (rx_frags[k].fAckMark == false); k--)

- In the event that *frag_ack* needs to be updated, *k > frag_ack:*
  - The value of *frag_ack* is updated to *k*.
  - An ARQ-ACK to LPDU with fragment counter *frag_ack* is generated.
  - At the time the last LPDU of the NPDU is acknowledged, *frag_ack == frag_last*, the NPDU is reassembled.

    If the reassembling is successful, the NPDU is forwarded to the upper layers. Otherwise, (i.e., bad size or CRC check) the NPDU is dropped.

– ***Next NPDU in flow received:*** The context is released when the next NPDU of the same flow is received.

8.6.1.2 Encapsulation

D018-COM-FUN-0780

The forward link encapsulation scheme shall limit the size of the L2 data units (LPDU), including payload and headers, to the payload size of the most robust FL MODCOD for data services using the ARQ protocol.

Note: Payload sizes for extended and non-extended FWD_DD fields and for the different FL MODCODs are provided in D018-COM-ITF-0920.

D018-COM-FUN-0790

The UT shall process LPDUs with multicast/broadcast L2 destination address either for data coming from GSE(s) assigned during logon/handover or for data without a GSE ID in the encapsulation header.

D018-COM-FUN-0791

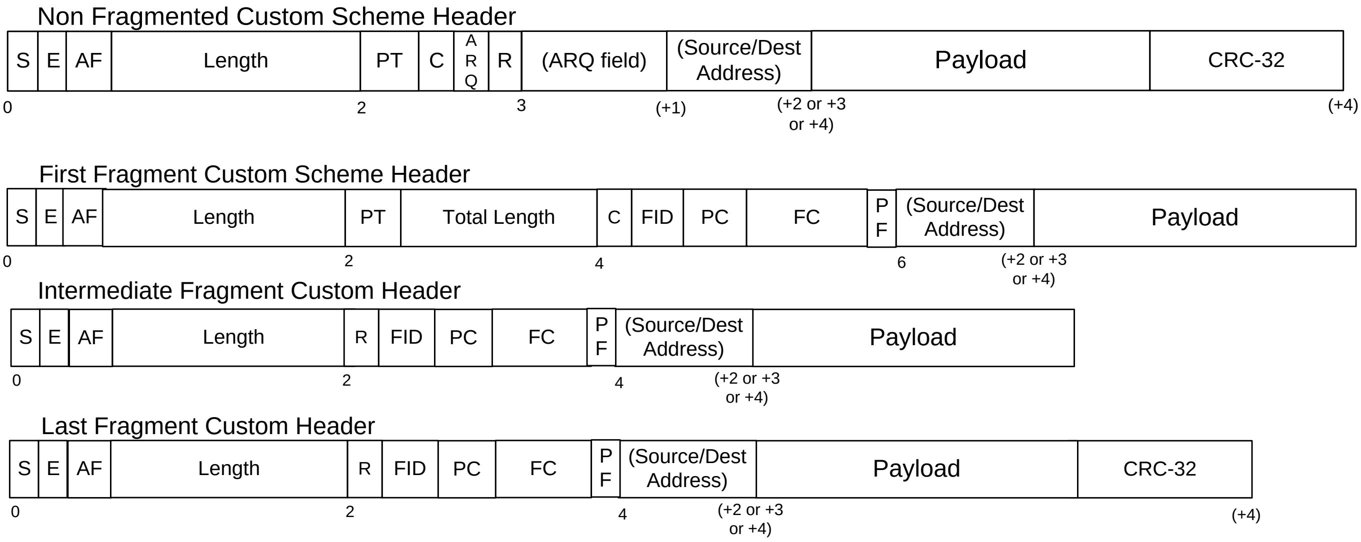
If several GSEs share a FWD link carrier, then transmitted multicast/broadcast LPDUs shall include the GSE ID in the encapsulation header.

D018-COM-FUN-0820

The forward link encapsulation scheme requires that a PPDU payload always starts with the encapsulation header of the first LPDU. Therefore, LPDU fragmentation between PPDU payloads is not allowed.

D018-COM-FUN-0830

The forward link encapsulation scheme used shall comply with the following:



where

Field	Length	Description
S and **E** flags: Start and End bits	1 bit each	These fields indicate if the packet is non-fragmented (11), or if it is the first fragment (10), the last fragment (01) or an intermediate fragment (00) of a fragmented packet. S and E bits set to 0 also permits signal padding when Length value field (see below) is set to 0 as well, meaning that the rest of the physical payload is composed of padding bits (0 bits). This padding signalling is required only when the physical payload remaining space after the last inserted fragment is longer or equal to 4 bytes; otherwise the rest of the L1 payload is composed of padding bits by default.
AF field: Address Format	2 bits	This field indicates the address format that will be contained in the encapsulation header. It can take the following values: **00**: No address field/address format re-use (re-use the address field of the previous LPDU in the same L1 payload) **01**: 2 byte address field: UT destination address **10**: 3 byte address field: 2 byte destination UT address + 1 byte source GS element address **11**: 4 byte address field: 3 byte destination UT ICAO address + 1 byte source GS element address
Length field	12 bits	This field indicates the length of the L2 payload size for the current fragment (for fragmented packet) or the total length of the payload (for non-fragmented packet). This length enables physical payload sizes up to 4099 bytes.

Total Length field	13 bits	This field indicates the total length of the L3 payload when fragmented. This length enables NPDU sizes up to 8191 bytes.
PT field: Payload Type	3 bits	This field indicates the type of the contained L2 payload. PT field can take the following values: **000**: This value of the PT field indicates that an ATN/IPS NPDU is encapsulated in the current LPDU (without header compression). **001**: This value of the PT field indicates that an ATN/OSI NPDU is encapsulated in the current LPDU. The ATN/OSI protocol self identifies whether the packet is compressed or not. **010**: This value of the PT field indicates that signalling information is encapsulated in the current LPDU. **011**: This value of the PT field indicates that an ATN/IPS header compressed NPDU is encapsulated in the current LPDU. The other values are reserved for future use.
C flag: CRC Presence	1 bits	This field indicates if a 4 byte CRC field is present as a trailer in the current packet (when un-fragmented) or in the last fragment (when fragmented).
ARQ flag	1 bits	This field indicates whether the ARQ support is required (1) for the LSDU transmission, or not (0). For un-fragmented LSDUs, the transmitter shall add the FID (4 bits) and PC (4 bits) fields to the encapsulation header whenever this flag is set to ‘1’.
R field: Reserved	3 bits/1 bit	This field is reserved for future use. Shall be set to 0.
Source/Dest Address field	optional field of 2, 3 or 4 bytes	This field is optional and contains the source and destination addresses as indicated by the Address Format (AF) field.
FID field: Flow ID	0 / 4 bits	This information is used by the ARQ process and/or by the re-assembly process for fragmented LSDUs. It shall comply with the Flow Identifier field of the ARQ header defined in §8.6.1.1.1. The presence of this field is mandatory for LPDUs associated with fragmented LSDUs. Otherwise, its presence depends on the value of the ARQ flag.
PC field: Packet Count	0 / 4 bits	This field identifies the LSDU currently sent. It is incremented for each LSDU sent toward a specific destination, using an independent counter for each flow ID. This field is used by ARQ process and/or by re-assembly process when the packet is fragmented. It shall comply with the Packet Counter field of the ARQ header defined in §8.6.1.1.1. The presence of this field in the encapsulation header is mandatory for LPDUs associated with fragmented LSDUs. Otherwise, its presence depends on the value of the ARQ flag.
FC field: Fragment Count	0 / 6 bits	This field is used as a counter for the fragments making up an LSDU. This field is used by ARQ process and/or by re-assembly process when the LSDU is fragmented. It shall comply with the Fragment Counter field of the ARQ header defined in §8.6.1.1.1. The presence of this field in the encapsulation header is mandatory and restricted to LPDUs associated with fragmented LSDUs.

P/F flag: Poll/Final bit	0 / 1 bit	This field is used to signal whether the fragment needs to be acknowledged or not by the ARQ process: P/F flag set to “1” means the fragment requires acknowledgement, P/F flag set to “0” means the fragment acknowledgement is not required.
CRC-32 field	4 bytes	When indicated by the C field, a trailer is added at the end of the last fragment when fragmented or after the payload when un-fragmented with a 4-byte CRC. The CRC-32 is computed as indicated in D018-COM-FUN-1070 and D018-COM-FUN-1080, considering the PNPDU as input stream.

8.6.1.3 Security

Refer to 8.3.

8.6.2 Physical layer specification

The following section specifies the physical layer of the user plane FWD link at 160 kbaud.

Note: refer to section 12 for physical layer specification for the FWD link at 16 kbaud.

8.6.2.1 Burst types

D018-COM-FUN-0840

The User plane FWD link shall support the following burst types:

- FCH burst

D018-COM-FUN-0850

The FCH burst shall support the following MODCODs:

- QPSK 1/4, 1/3, 1/2, 2/3
- 8-PSK 1/2, 2/3
- 16-APSK 2/3

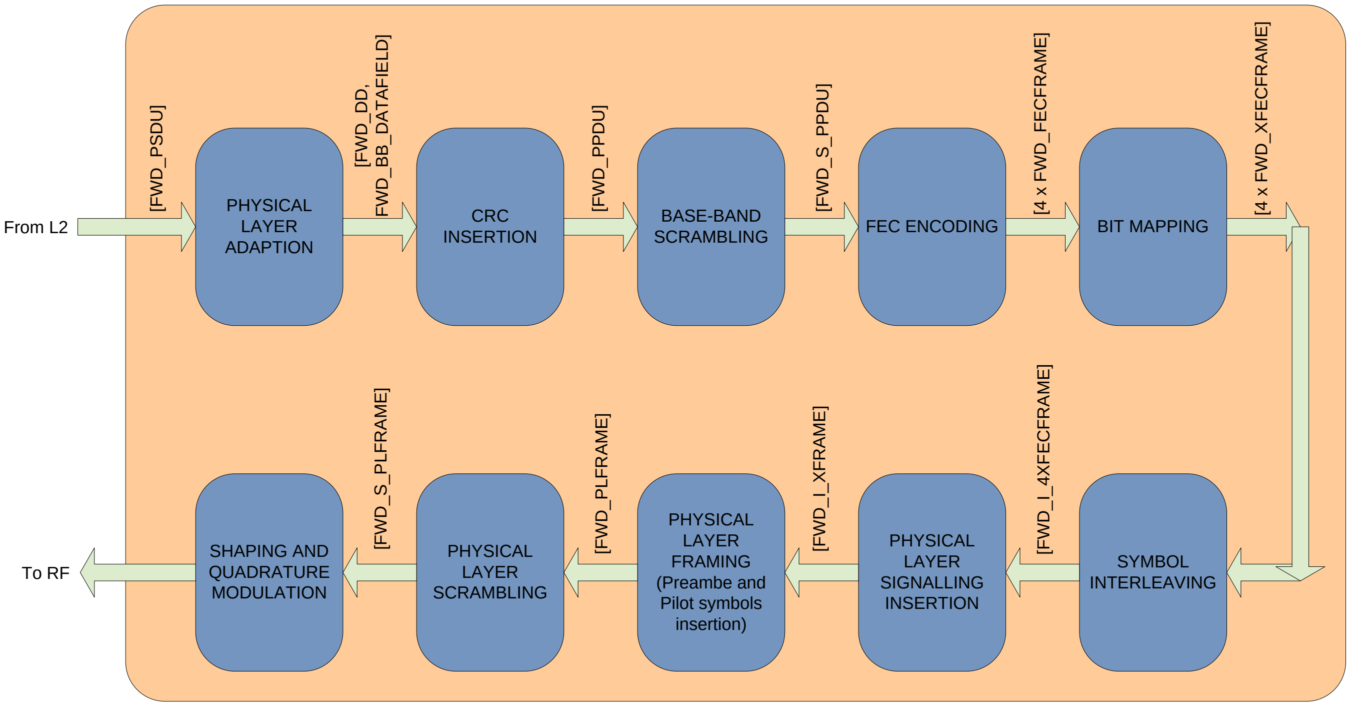
8.6.2.2 Burst waveform generation

D018-COM-FUN-0860

The burst waveform generation shall be applied to the FWD_PSDU and shall be composed of a sequence of functional modules as represented in the following figure. The functional modules are:

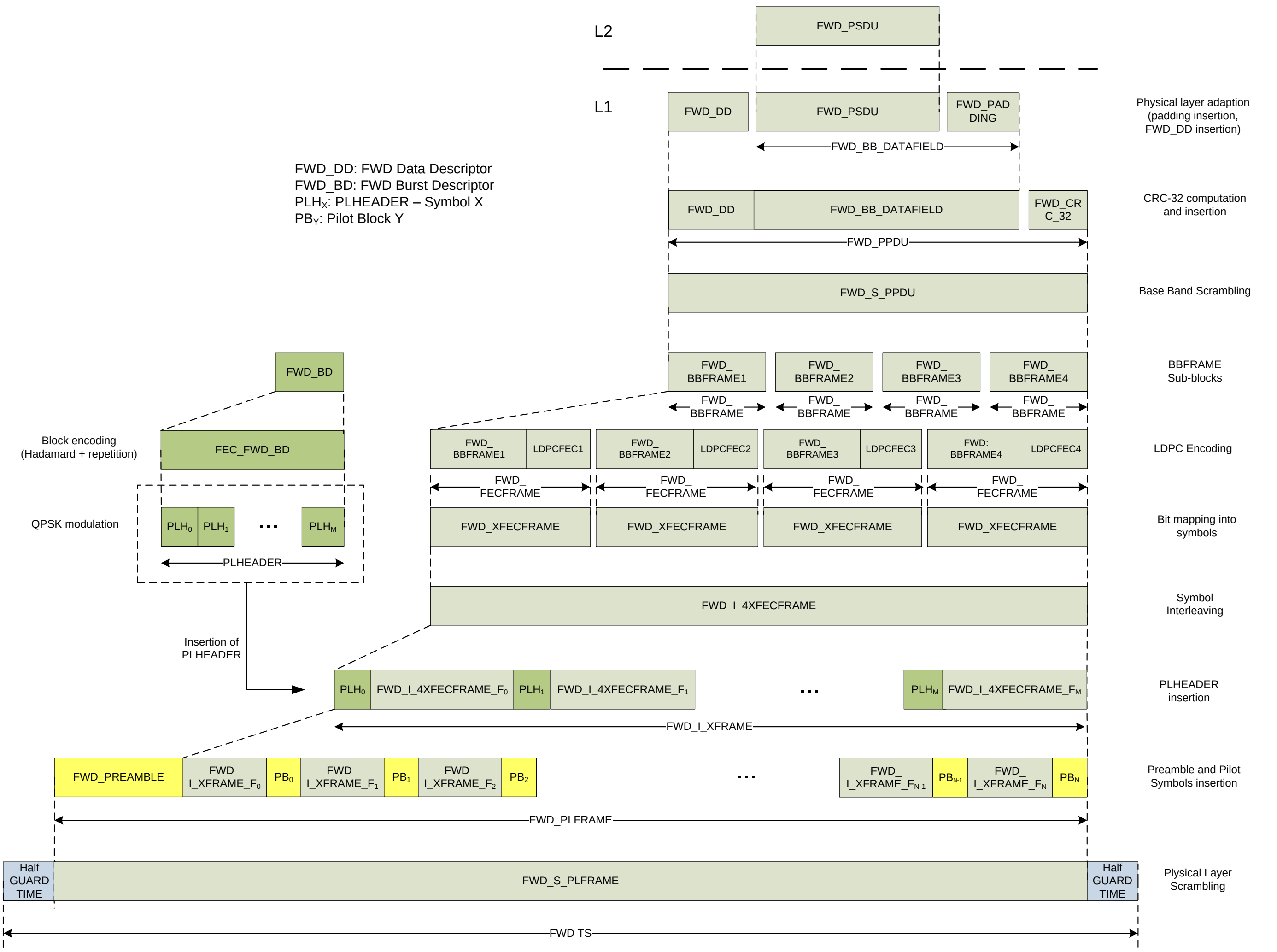
- Physical Layer Adaptation
- CRC insertion

- Base Band scrambling
- FEC Encoding, which includes the inner encoding and the bit interleaver
- Bit Mapping into Constellation
- Symbol Interleaving
- Physical Layer Signalling Generation and Insertion
- Physical Layer Framing
- Physical Layer Scrambling
- Base-band Pulse Shaping and Quadrature Modulation



D018-COM-FUN-0870

FCH bursts shall follow the burst format presented in the following figure:



8.6.2.3 *Physical Layer Adaptation*

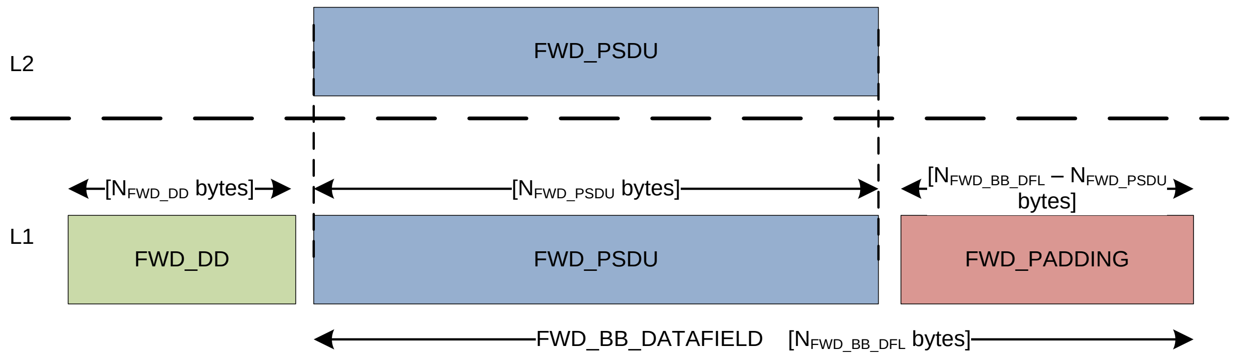
D018-COM-FUN-0880

The Physical Layer Adaptation module shall perform:

- Interface with Layer 2
- Padding insertion
- FWD_DD (FWD Data Descriptor) insertion

D018-COM-FUN-0890

The input stream of the Physical Layer Adaptation module shall be a FWD_PSDU and the output stream a FWD_DD Header followed by a FWD_BB_DATAFIELD, as detailed in the following figure.



D018-COM-FUN-0900

A single FWD_PSDU shall be mapped onto a single FCH burst.

8.6.2.3.1 Interface with Layer 2

D018-COM-ITF-0910

The Physical Layer Adaptation module shall map FWD_PSDU of up to 4 types of layer 2 encapsulation protocols:

- FL Encapsulation, as defined in D018-COM-FUN-0830
- Protocol 1 (RFU)
- Protocol 2 (RFU)
- Protocol 3 (RFU)

D018-COM-ITF-0920

The FWD_PSDU size (in bytes) shall be variable in the range. The maximum PSDU size (Max. N_{FWD_PSDU}) depends on the selected MODCOD and on the FWD_DD size (N_{FWD_DD}) as detailed in the following table:

Mode	Max. N_{FWD_PSDU} [Maximum PSDU size (bytes)]	
	Non-extended FWD_DD (N_{FWD_DD} = 3 bytes)	Extended FWD_DD (N_{FWD_DD} = 8 bytes)
QPSK 1/4	761	756
QPSK 1/3	1017	1012
QPSK 1/2	1529	1524
QPSK 2/3	2041	2036
8-PSK 1/2	2297	2292
8-PSK 2/3	3065	3060
16-APSK 2/3	4089	4084

Note: Maximum PSDU size (Max. N_{FWD_PSDU}) corresponds with the FWD_BB_DATAFIELD size ($N_{FWD\_BB\_DFL}$)

8.6.2.3.2 Padding insertion

D018-COM-FUN-0930

($N_{FWD\_BB\_DFL}$ - N_{FWD_PSDU}) bytes shall be appended after the FWD_PSDU according to figure in D018-COM-FUN-1000. The content of the padding bytes is "0x00".

8.6.2.3.3 FWD_DD (FWD Data Descriptor) insertion

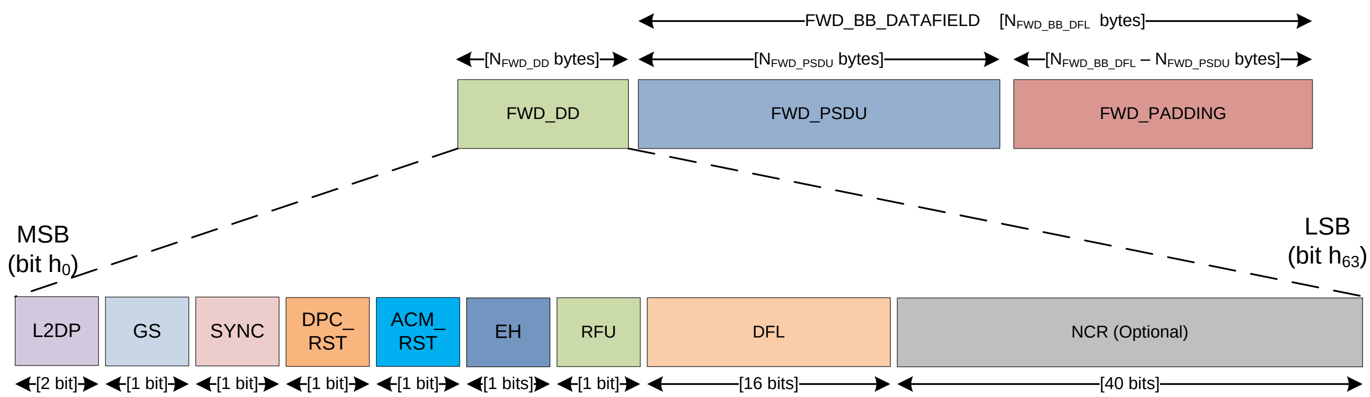
D018-COM-FUN-1000

A header (FWD_DD) of N_{FWD_DD}= 3 bytes ($h = h_0, \ldots, h_{23}$) - Non-extended FWD_DD - or N_{FWD_DD}= 8 bytes ($h = h_0, \ldots, h_{63}$) - Extended FWD_DD - shall be inserted in front of the FWD_BB_DATAFIELD according to the figure below.

The FWD_DD header contains the following fields:

Bits	Field	Field size	Description
h_0-h_1	L2DP	2 bits	It indicates the L2 Data Protocol (L2DP) used in the FWD_PSDU.
h_2	GS	1 bit	It indicates the GS Source (NCC or GES).

h_3	SYNC	1 bit	It indicates whether the burst is synchronous with respect to the NCC.
h_4	DPC_RST	1 bit	It indicates the need to reset the UT Doppler pre-compensation mechanism (this field is used in toggle mode).
h_5	ACM_RST	1 bit	It indicates the need to reset the UT ACM mechanism (this field is used in toggle mode).
h_6	EH	1 bit	It indicates whether the FWD_DD is extended or not.
h_7	Reserved for future use	1 bit	---
h_8-h_{23}	DLF	16 bits	It contains the FWD_PSDU length (Data Length Field - DFL) in bytes. Range [0-65535].
h_{24}-h_{63}	NCR	40 bits	Optional field (only applicable for Extended FWD_DD). It carries the value of the NCR.



D018-COM-FUN-1010

The L2DP field shall indicate which L2 entity (encapsulation protocol) is the client of the burst. It is a 2-bit field coded as detailed in the following table:

bit h_0- bit h_1	Description
00	FL Encapsulation
01	RFU
10	RFU

11	RFU

D018-COM-FUN-1020

The GS field shall indicate which Ground Segment (NCC or GES) entity has transmitted the burst. The GS field is a 1-bit field coded as detailed in the following table.

bit h_2	Description
0	NCC
1	GES

D018-COM-FUN-1030

The SYNC field shall indicate whether the burst is synchronous with respect to the NCC transmissions. SYNC field is a 1-bit field coded as detailed in the following table.

bit h_3	Description
0	Burst “not synchronous” w.r.t. NCC transmissions
1	Burst “synchronous” w.r.t. NCC transmissions

The criterion to determine whether a burst is synchronous with respect to the NCC transmissions shall be as follows:

- If the burst is transmitted by the NCC, SYNC flag is always set to "1".
- If the burst is transmitted by a GES, the SYNC flag is set to "1" when the GES is in the synchronisation maintenance stage.

Note: the information about whether the burst is synchronous or not with regard to the NCC transmissions shall be used by the UT in order to decide whether the burst can be used to compute the Doppler pre-compensation or not.

D018-COM-FUN-1031

The DPC_RST (Doppler Pre-Compensation Reset) field (bit h_4) shall be used in toggle mode to signal the need to reset the UT Doppler pre-compensation mechanism (every time the DPC_RST bit changes its state, the UT Doppler pre-compensation mechanism is reset).

D018-COM-FUN-1032

The ACM_RST (ACM Reset) field (bit h_5) shall be used in toggle mode to signal the need to reset the ACM mechanism (every time the ACM_RST bit changes its state, the ACM mechanism is reset).

D018-COM-FUN-1033

The EH (Extended Header) field shall indicate whether the FWD_DD is extended or not.

bit h_6	Description
0	Non-extended FWD_DD (N_{FWD_DD} = 3 bytes)
1	Extended FWD_DD (N_{FWD_DD} = 8 bytes)

D018-COM-FUN-1034

The NCR field shall carry the value of the Clock counter when the last symbol of the preamble is transmitted. The NCR field is only be inserted when the EH field is set to “1”.

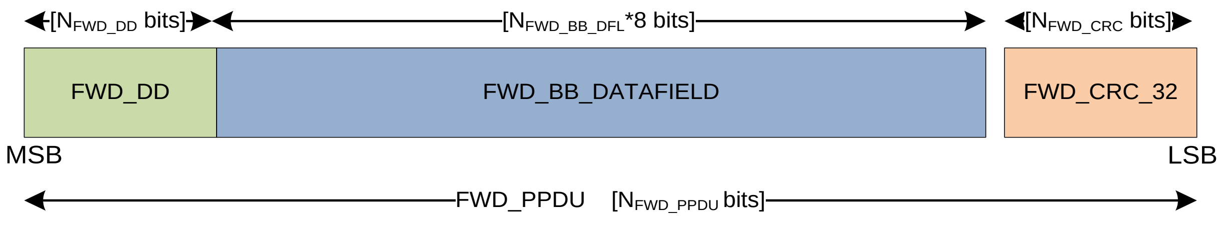
8.6.2.4 CRC insertion

D018-COM-FUN-1040

The FWD CRC insertion module shall compute the CRC parity bits in order to detect erroneous packets at the receiver end and to provide a packet quality indicator.

D018-COM-FUN-1050

The input stream of the FWD CRC Insertion module shall be a FWD_DD header followed by a FWD_BB_DATAFIELD and the output stream a FWD_PPDU, as illustrated in the following figure.



D018-COM-FUN-1060

The contents of the FWD_CRC_32 field shall be the result of processing

- the FWD_DD field and
- the FWD_BB_DATAFIELD field

by a systematic 32-bit CRC encoder and appended after the FWD_BB_DATAFIELD, as illustrated in the figure for D018-COM-FUN-1050 above.

D018-COM-FUN-1070

The CRC parity bits (FWD_CRC_32 field) shall be computed as the remainder of the division of the input stream (FWD_DD and FWD_BB_DATAFIELD) by the generator polynomial

$$G(X) = X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^{8} + X^{7} + X^{5} + X^{4} + X^{2} + X + 1$$

$$CRC = R(X) = [X^{32} \cdot M(X)]mod(G(X))$$

where,

- all the arithmetic is in modulo 2
- M(X) is the input stream to be processed by the systematic 32-bit CRC encoder expressed as a polynomial with binary coefficients:

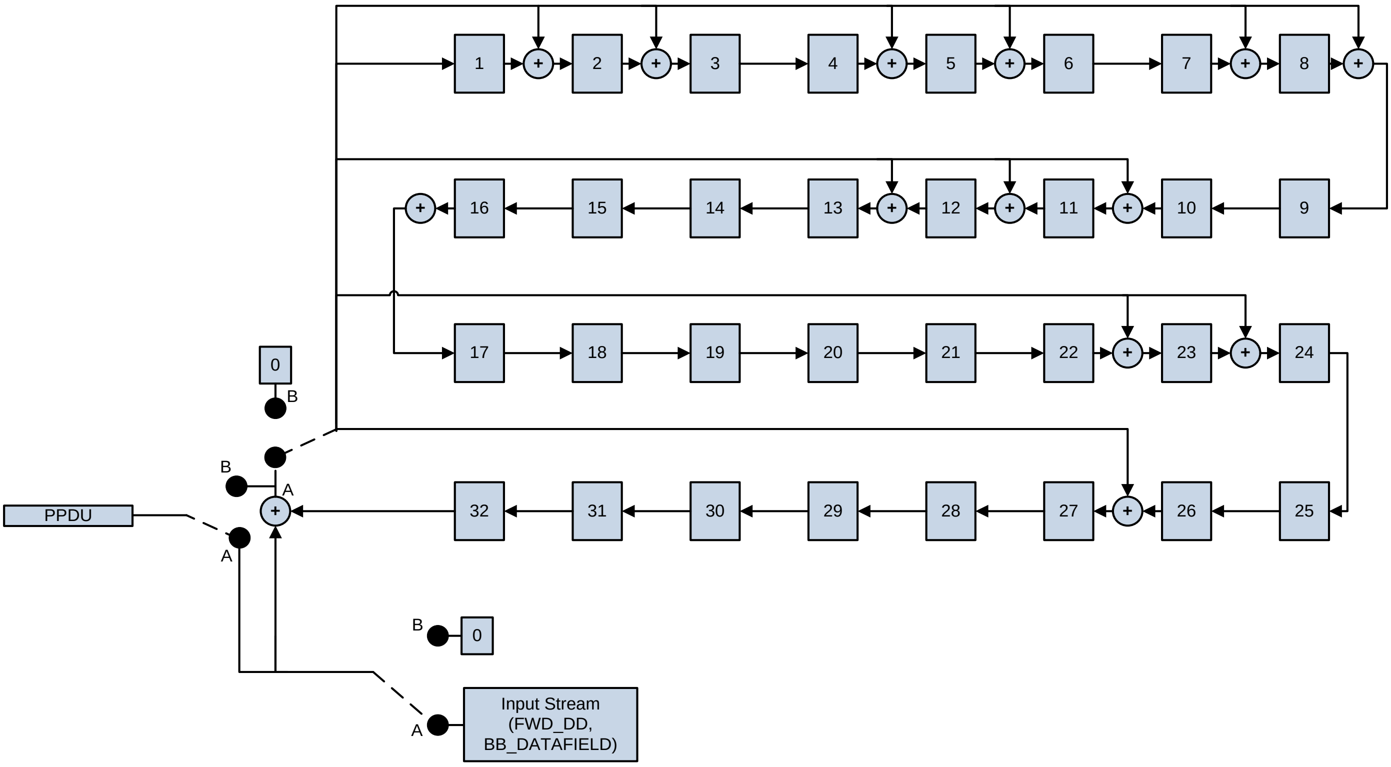
$$M(X) = m_{n-1}X^{n-1} + \cdots + m_1X^1 + m_0$$

D018-COM-FUN-1080

The contents of the FWD_CRC_32 field shall be equal to the value computed by the following procedures and the shift register structure shown in the figure below:

1. The shift register cells are initialized to 1.
2. All the switches are set in the A position.
3. The shift register is clocked a number of times equal to the number of input bits (FWD_DD + FWD_BB_DATAFIELD). The MSB of the FWD_DD is the first bit to be inserted in the shift register.
4. Once the last input bit has been inserted in the shift register (LSB of FWD_BB_DATAFIELD), the switches are set to position B, forcing the inputs to the shift register to be 0.
5. The shift register is clocked an additional number of times equal to the number of CRC parity bits (i.e., 32). The 32 additional bits are the CRC parity bits (FWD_CRC_32 field).
6. The CRC parity bits are transmitted in the order of generation (MSB first).

All the addition operations are performed in GF(2)



8.6.2.5 Base-band scrambling

D018-COM-FUN-1090

The Base-band scrambling module shall add a binary pseudo-noise sequence to the input data stream in order to randomise the binary transitions in the output data stream.

D018-COM-FUN-1100

The input stream of the Base-band scrambling module shall be a FWD_PPDU and the output stream a FWD_S_PPDU (FWD Scrambled PPDU).

Note: the length of the input and output data streams (FWD_PPDU and FWD_S_PPDU) is the same, as the Base-band scrambling does not add redundancy, i.e., N_{FWD_PPDU} *bits =* $N_{FWD\_S\_PPDU}$ *bits.*

D018-COM-FUN-1110

The complete FWD_PPDU shall be randomized using the Base-band scrambling. The binary pseudo-noise sequence is synchronous with the FWD_PPDU, starting with the MSB of the FWD_PPDU.

D018-COM-FUN-1120

The binary pseudo-random sequence (scrambling sequence) shall be generated by a linear feedback shift register with connection polynomial

$$h(D) = 1 + D^{14} + D^{15}$$

D018-COM-FUN-1130

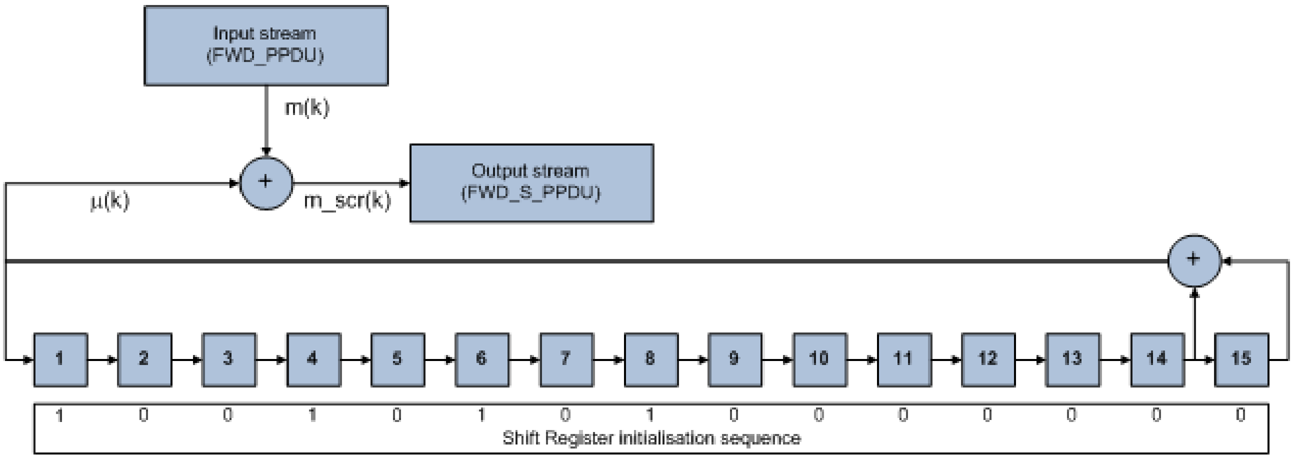
The FWD_S_PPDU shall be generated by adding in modulo 2 the pseudo-random sequence with the connection polynomial specified in D018-COM-FUN-1120 to the input data stream (FWD_PPDU), as shown in the figure below. The procedure to generate the FWD_S_PPDU from a FWD_PPDU is as follows:

1. At the start of every FWD_PPDU, the contents of the shift register cells are initialised with the following sequence (1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0), as indicated in the figure below.
2. The first value of the pseudo-noise sequence is $\mu(0) = 0$, being the modulo-2 sum of the two last elements in the shift register after initialization. For each input bit of the FWD_PPDU frame, the shift register is clocked once to generate the next element of the pseudo-noise sequence.
3. After each clocking of the shift register, the output bit of the FWD_S_PPDU frame is computed according to the rule

   $$m\_scr(k) = [m(k) + \mu(k)]mod2$$

   where k represents the k^{th} clocking of the shift register.

The shift register is clocked a number of times equal to the number of inputs bits of FWD_PPDU frame (N_{FWD_PPDU}). The MSB of the FWD_PPDU is randomised first.



8.6.2.6 FEC Encoding

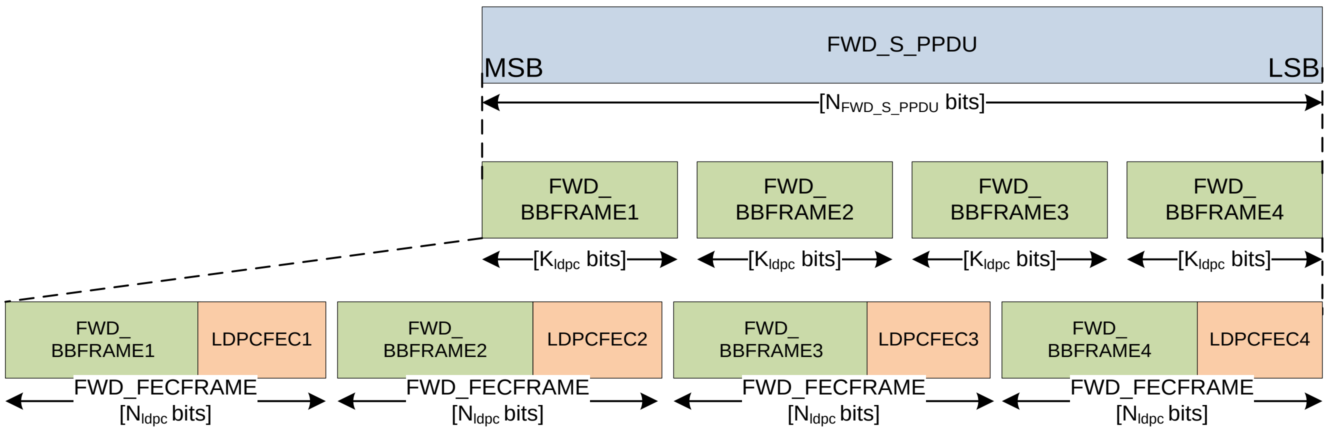
D018-COM-FUN-1140

The FEC Encoding module shall perform the following functions:

- Inner channel coding (IRA LDPC)
- Bit interleaving

D018-COM-FUN-1150

The input stream of the FEC encoding module shall be a FWD_S_PPDU and the output stream 4 FWD_FECFRAME, as illustrated in the following figure.



D018-COM-FUN-1160

The FWD_S_PPDU ($N_{FWD\_S\_PPDU}$ bits) shall be divided in 4 FWD_BBFRAME of K_{ldpc} bits each starting from the MSB (i.e., FWD_BBFRAME1 contains the MSB of FWD_S_PPDU). K_{ldpc} depends on the supported MODCOD types (Mode). The coding parameters are provided in the following table.

MODCOD Id	MODCOD	$N_{FWD\_S\_PPDU}$ (bits)	K_{ldpc} (bits)	N_{ldpc} (bits)
MODCOD0	QPSK 1/4	6144 bits	1536 bits	6144 bits
MODCOD1	QPSK 1/3	8192 bits	2048 bits	6144 bits
MODCOD2	QPSK 1/2	12288 bits	3072 bits	6144 bits
MODCOD3	QPSK 2/3	16384 bits	4096 bits	6144 bits
MODCOD4	8-PSK 1/2	18432 bits	4608 bits	9216 bits
MODCOD5	8-PSK 2/3	24576 bits	6144 bits	9216 bits
MODCOD6	16-APSK 2/3	32768 bits	8192 bits	12288 bits

D018-COM-FUN-1170

The 4 FWD_BBFRAME belonging to the same FWD_S_PPDU shall be encoded with the same MODCOD.

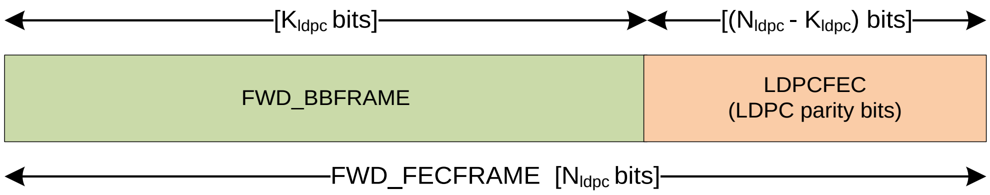
8.6.2.6.1 Inner Encoding (LDPC)

D018-COM-FUN-1180

Each $FWD\_BBFRAME_i$ (K_{ldpc} bits) shall be encoded using IRA LDPC codes to generate a $FWD\_FECFRAME_i$ (N_{ldpc} bits).

D018-COM-FUN-1190

LDPC parity (LDPCFEC) bits shall be appended after the FWD_BBFRAME, as illustrated in the following figure.



D018-COM-FUN-1200

The LDPC encoder shall systematically encode a FWD_BBFRAME block of size K_{ldpc}, $\boldsymbol{i} = (i_0, i_1, i_2, i_3, \ldots, i_{kldpc-3}, i_{kldpc-2}, i_{kldpc-1})$ onto a codeword (FWD_FECFRAME) of size N_{ldpc}, $\boldsymbol{c}_{ldpc} = (i_0, i_1, i_2, \ldots, i_{kldpc-2}, i_{kldpc-1}, p_0, p_1, \ldots, p_{nldpc-kldpc-2}, p_{nldpc-kldpc-1})$.

The procedure to determine the N_{ldpc} - K_{ldpc} parity bits $\boldsymbol{p} = (p_0, p_1, \ldots, p_{nldpc-kldpc-2}, p_{nldpc-kldpc-1})$ is as follows:

1. Initialise $p_0 = p_1 = \ldots = p_{nldpc-kldpc-2} = p_{nldpc-kldpc-1} = 0$
2. Accumulate the first information bit i_0 at parity bit addresses specified in the first row of Table 13-1 through Table 13-7 in section 13. For example, for code rate = 1/2 and Kldpc = 3072 bits (Table 13-2) (*note: all additions are in GF(2)*).

$$p_{1001} = p_{1001} \oplus i_0$$
$$p_{1397} = p_{1397} \oplus i_0$$
$$p_{1561} = p_{1561} \oplus i_0$$
$$p_{2604} = p_{2604} \oplus i_0$$
$$p_{2768} = p_{2768} \oplus i_0$$

3. For the next c-1 information bits (FWD_BBFRAME bits) i_m, *m=1, 2, ..., c-1* accumulate i_m at parity bit addresses $\{x + (m\ mod c) \cdot q\} mod(N_{ldpc} - K_{ldpc})$

   where x denotes the address of the parity bit of the accumulator corresponding to the first bit i_0. c and q values are specified in the following table.

Block Size (K_{ldpc}), code rate (r)	ModCod Id	q/c values
K_{ldpc} = 1536 bits; r=1/4	0	36/128
K_{ldpc} = 2048 bits; r= 1/3	1	32/128
K_{ldpc} = 3072 bits; r=1/2	2	24/128
K_{ldpc} = 4096 bits; r=2/3	3	16/128
K_{ldpc} = 4608 bits; r=1/2	4	36/128
K_{ldpc} = 6144 bits; r=2/3	5	24/128
K_{ldpc} = 8192 bits; r=2/3	6	16/256

Continuing with the example (code rate = 1/2 and Kldpc = 3072 bits), *c = 128* and *q = 24*. So for the information bit i_1, the following operations are performed:

$$p_{1025} = p_{1025} \oplus i_1$$
$$p_{1421} = p_{1421} \oplus i_1$$
$$p_{1585} = p_{1585} \oplus i_1$$
$$p_{2628} = p_{2628} \oplus i_1$$
$$p_{2792} = p_{2792} \oplus i_1$$

4. For the $(c+1)^{th}$ information bit i_c, the addresses of the parity bit accumulators are given in the second row of Table 13-1 through Table 13-6. In a similar manner the addresses of the parity bit accumulators for the following c-1 information bits i_m, m=c+1, ..., 2c-1 are obtained using the formula $\{x + (m\ mod c) \cdot q\} mod(N_{ldpc} - K_{ldpc})$ where x denotes the address of the parity bit accumulator corresponding to the information bit i_c, i.e., the entries in the second row of Table 13-1 through Table 13-7.
5. In like fashion, for every group of c new information bits, a new row from Table 13-1 through Table 13-7 is used to find the addresses of the parity bit accumulators.
6. After all of the information bits are exhausted, the final parity bits are obtained as follows:
   a. sequentially perform the following operations starting with i = 1

   $$p_i = p_i \oplus p_{i-1}, \quad i = 1, 2, \ldots, N_{ldpc} - K_{ldpc} - 1$$

   b. final content of p_i, i = 0, 1, N_{ldpc}-K_{ldpc} -1 is equal to the parity bit p_i.

8.6.2.6.2 Bit interleaving (for 8-PSK and 16-APSK only)

D018-COM-FUN-1210

The output of the LDPC encoder and puncturing (FWD_FECFRAME) shall be bit interleaved using a block interleaver (row/column) with the parameters defined in the following table:

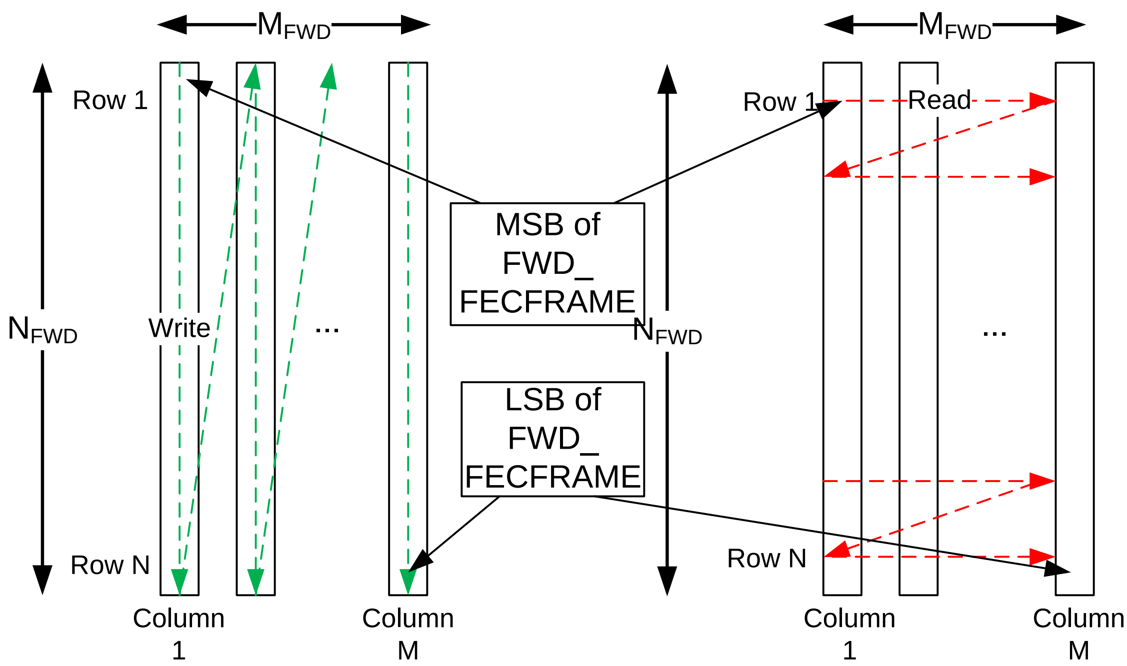
Modcod Id	**MODCOD**	**N_{FWD} (Rows)**	**M_{FWD} (Columns)**
MODCOD 0	QPSK 1/4	N.A.	N.A.
MODCOD 1	QPSK 1/3	N.A.	N.A.
MODCOD 2	QPSK 1/2	N.A.	N.A.
MODCOD 3	QPSK 2/3	N.A.	N.A.
MODCOD 4	8-PSK 1/2	3072	3
MODCOD 5	8-PSK 2/3	3072	3
MODCOD 6	16-APSK 2/3	3072	4
Note: – N stands for the number of rows – M stands for the number of columns			

D018-COM-FUN-1220

The bit interleaver depth shall be 1 FWD_FECFRAME (1 Code Word)

D018-COM-FUN-1230

On the transmitter end, the encoded bits shall be serially written into the interleaver column-wise and serially read out row-wise according to the figure below. MSB of FWD_FECFRAME is written and read out first.



D018-COM-FUN-1240

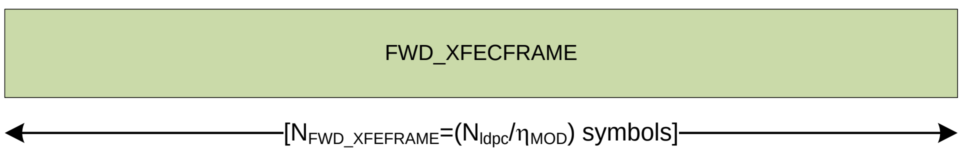
On the receiver end, the encoded bits shall be serially written in row-wise and serially read out column-wise. MSB of FWD_FECFRAME is written and read out first.

8.6.2.7 Bit mapping into constellation

The bit mapping module shall perform the mapping of the incoming bits into symbols (complex values) according to the specified constellation order.

D018-COM-FUN-1260

The input stream of the bit mapping module shall be a FWD_FECFRAME and the output stream a FWD_XFECFRAME (FWD compleX FECFRAME), as illustrated schematically below.



D018-COM-FUN-1270

Each FWD_FECFRAME shall be serial-to-parallel converted. Three parallel levels are supported (parallel level = η_{MOD})

- η_{MOD}=2 for QPSK
- η_{MOD}=3 for 8-PSK
- η_{MOD}=4 for 16-APSK

Then, each parallel sequence is mapped into constellation, generating a complex (I,Q) sequence (FWD_XFECFRAME) according to the modulation efficiency (η_{MOD}) as described in section 8.6.2.7.1 for η_{MOD}=2, section 8.6.2.7.2 for η_{MOD}=3 and section 8.6.2.7.3 for η_{MOD}=4.

D018-COM-FUN-1280

The number of symbols after the bit mapping shall be constant, as detailed in the following table.

Modcod Id	MODCOD	N_{ldpc} (bits) FWD_FECFRAME size	Mod. efficiency (η_{MOD})	$N_{FWD\_XFECFRAME}$ (symbols) FWD_XFECFRAME size
MODCOD0	QPSK 1/4	6144 bits	2	3072 symbols
MODCOD 1	QPSK 1/3	6144 bits	2	3072 symbols
MODCOD 2	QPSK 1/2	6144 bits	2	3072 symbols
MODCOD 3	QPSK 2/3	6144 bits	2	3072 symbols
MODCOD 4	8-PSK 1/2	9216 bits	3	3072 symbols
MODCOD 5	8-PSK 2/3	9216 bits	3	3072 symbols
MODCOD 6	16-APSK 2/3	12288 bits	4	3072 symbols

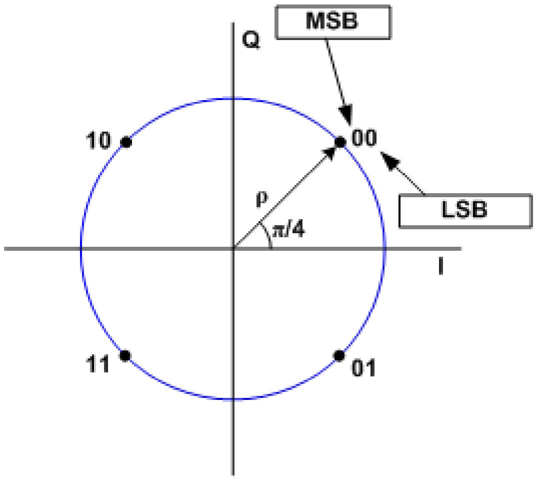
D018-COM-FUN-1290

The FWD_FECFRAMEs belonging to the same FWD_S_PPDU (4 FWD_FECFRAME) shall be modulated (bit mapped) using the same modulation format.

8.6.2.7.1 Bit mapping into QPSK constellation

D018-COM-FUN-1300

The bit mapping into QPSK constellation shall be compliant with the figure below. Two FWD_FECFRAME bits are mapped to a QPSK symbol.

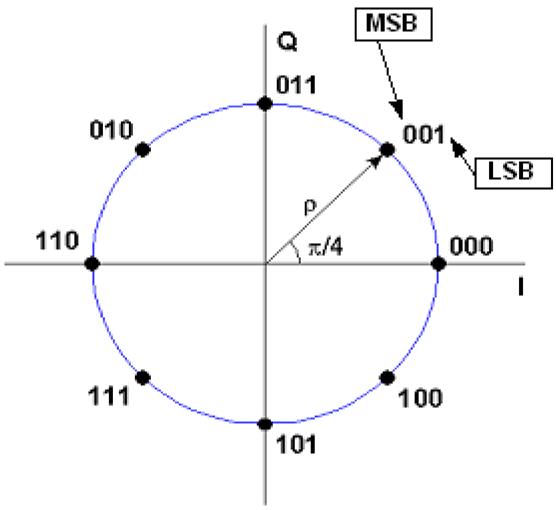


Note: the symbol energy is ρ^2.

8.6.2.7.2 Bit mapping into 8-PSK constellation

D018-COM-FUN-1310

The bit mapping into the 8-PSK constellation shall be compliant with the figure below. Three FWD_FECFRAME bits are mapped to a 8-PSK symbol.



Note: the symbol energy is ρ^2.

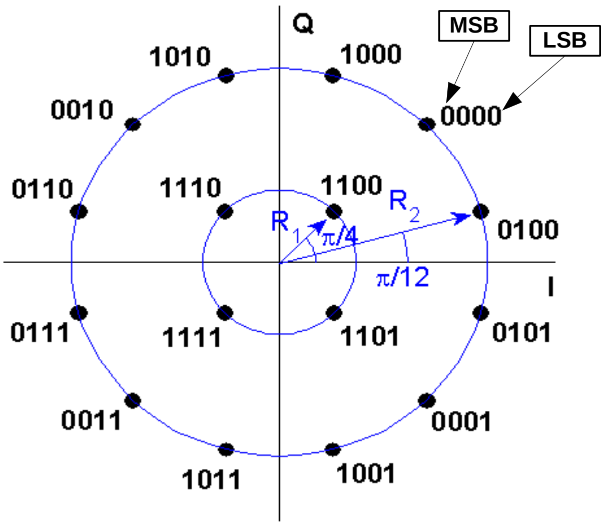
8.6.2.7.3 Bit mapping into 16-PSK constellation

D018-COM-FUN-1320

The 16-APSK modulation constellation shall be composed of two concentric rings of uniformly spaced 4 and 12-PSK points respectively in the inner ring of radius R_1 and the outer ring of radius R_2 (See D018-COM-FUN-1330).

D018-COM-FUN-1330

The bit mapping of the 16-PSK constellation shall be compliant with the figure below. Four FWD_FECFRAME bits are mapped to a 16-APSK symbol.



Note: the average symbol energy is $(4R_1^2+12R_2^2)/16$.

D018-COM-FUN-1340

The ratio of the outer circle radius to the inner circle radius shall comply with the following table.

Code rate	Modulation/coding spectral efficiency	γ
2/3	2.66	3.15

Note: the ratio between the outer and inner circle depends on the code rate.

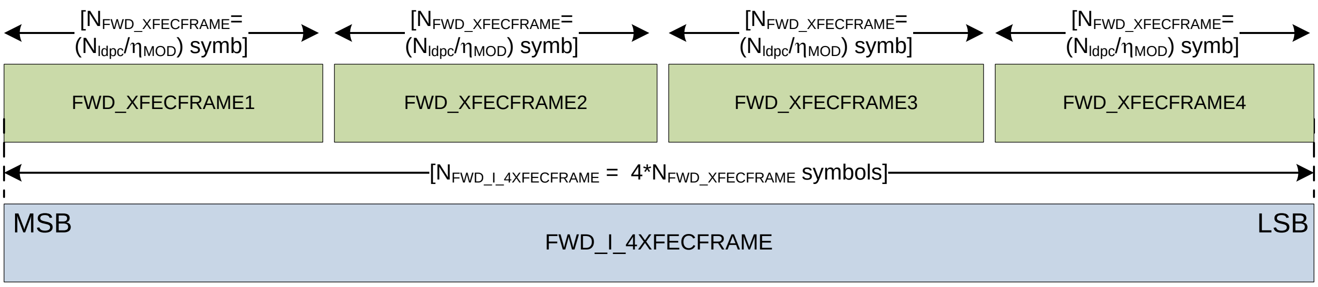
8.6.2.8 Symbol interleaving

D018-COM-FUN-1350

The Symbol Interleaving module shall interleave the symbols of 4 FWD_XFECFRAMES in order to exploit the time diversity.

D018-COM-FUN-1360

The input stream of the Symbol Interleaving module shall be 4 FWD_XFECFRAME ($FWD\_XFECFRAME_i$, i=1...4) and the output a FWD_I_4XFECFRAME.



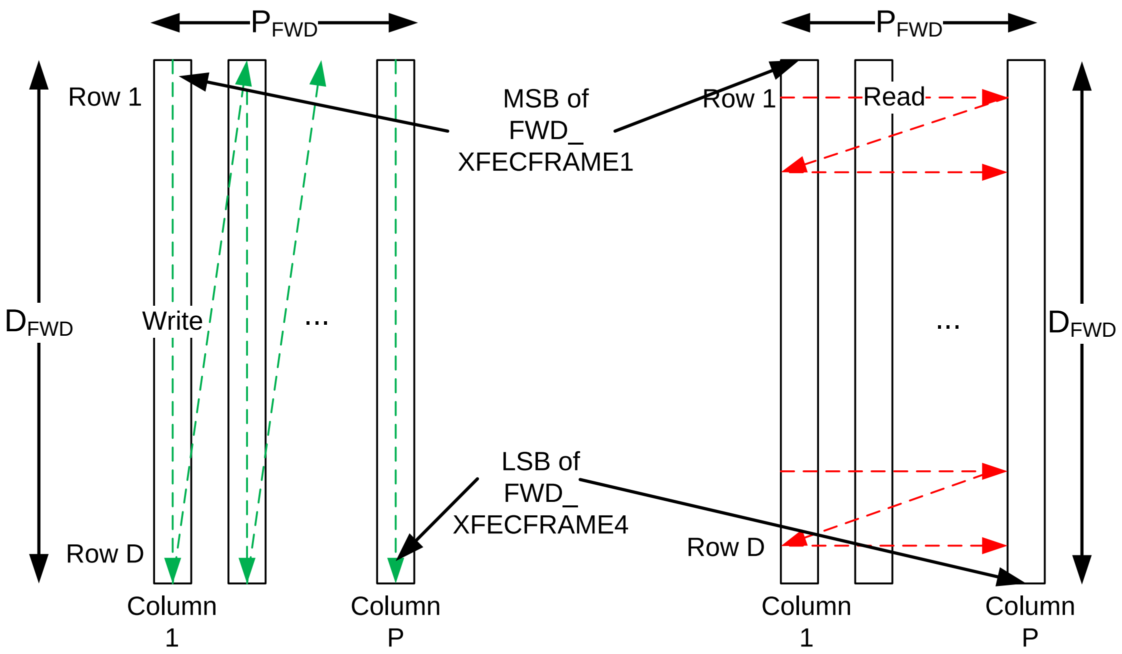
D018-COM-FUN-1370

4 FWD_XFECFRAMES shall be symbol interleaved using a block interleaver (row/column) with the parameters defined in the following table.

Modcod Id	**MODCOD**	D_{FWD} **(Number of rows)**	P_{FWD} **(Number of colums)**	$N_{FWD\_I\_4XFECFRAME}$ **(symbols)**
MODCOD 0	QPSK 1/4	192	64	12288 symbols
MODCOD 1	QPSK 1/3	192	64	12288 symbols
MODCOD 2	QPSK 1/2	192	64	12288 symbols
MODCOD 3	QPSK 2/3	192	64	12288 symbols
MODCOD 4	8-PSK 1/2	192	64	12288 symbols
MODCOD 5	8-PSK 2/3	192	64	12288 symbols
MODCOD 6	16-APSK 2/3	192	64	12288 symbols
Note: – D_{FWD} stands for the number of rows – P_{FWD} stands for the number of columns				

D018-COM-FUN-1380

On the transmitter end, the symbols shall be serially written in column-wise (starting from the MSB of FWD_XFECFRAME1 and ending with the LSB of FWD_XFECFRAME4) and serially read out row-wise according to the following figure.



D018-COM-FUN-1390

On the receiver end, the symbols shall be serially written in row-wise and serially read out column-wise.

8.6.2.9 Physical Layer Signalling Generation and Insertion

D018-COM-FUN-1400

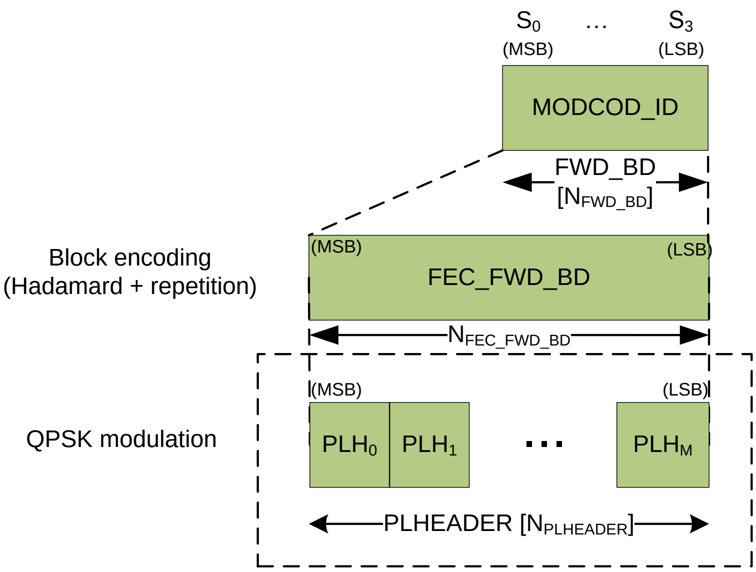
The input stream of the Physical Layer Signalling Generation and Insertion shall be a FWD_I_4XFECFRAME and the output stream a FWD_I_XFRAME.

8.6.2.9.1 Physical Layer Signalling Generation

D018-COM-FUN-1410

The FWD_BD (FWD Burst Descriptor) field shall break into:

- The MODCOD_ID field (4 bits)



D018-COM-FUN-1420

The MODCOD_ID field indicates which MODCOD has been used for transmitting the FWD_I_4XFECFRAME. The MODCOD_ID shall be coded on $N_{FWD\_BD} = 4$ bits s = (s_0, s_1, s_2, s_3) according to the following table.

Mode	MODCOD_ID			
	bit s_0	bit s_1	bit s_2	bit s_3
QPSK 1/4	0	0	0	0
QPSK 1/3	0	0	0	1
QPSK 1/2	0	0	1	0
QPSK 2/3	0	0	1	1
8-PSK 1/2	0	1	0	0
8-PSK 2/3	0	1	0	1
16-APSK 2/3	0	1	1	0

Note: The other possible combinations of (s_0, s_1, s_2, s_3) are Reserved for Future Use (RFU).

D018-COM-FUN-1430

The FEC_FWD_BD field shall be constructed using the Hadamard code (8, 4, 4) concatenated with code repetition (16 times). The procedure to generate FEC_FWD_BD is as follows:

1. The FWD_BD field s = (s_0, s_1, s_2, s_3) is first encoded using the Hadamard code of length 8 (8, 4, 4) with the following generator matrix:

$$G = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

The 8-bit code-word (c_{HAD}) produced is given by

$$\boldsymbol{c_{HAD}} = (s_0, s_1, s_2, s_3)G = (c_{HAD\_0}, c_{HAD\_1}, c_{HAD\_2}, c_{HAD\_3}, c_{HAD\_4}, c_{HAD\_5}, c_{HAD\_6}, c_{HAD\_7})$$

where the matrix multiplication is performed in GF(2). The input string (s) and the output string (c) are horizontal vectors.

2. The 8-bit code-word c_{HAD} is repeated 16 times to get the FEC_FWD_BD bit pattern ($N_{FEC\_FWD\_BD}$ bits)

$$\boldsymbol{l} = (l_0, l_1, l_2, \ldots, l_{125}, l_{126}, l_{127}), \quad with \quad l_m = c_{HAD\_m(mod 8)}$$

D018-COM-FUN-1440

FEC_FWD_BD, represented by the binary sequence l = (l_0, l_1, ..., l_{127}), shall be modulated into $N_{PLHEADER}$ = 64 QPSK symbols **PLH =** (PLH_1, PLH_2, ..., PLH_{64}). Bit mapping into QPSK modulation is performed according to D018-COM-FUN-1300 above.

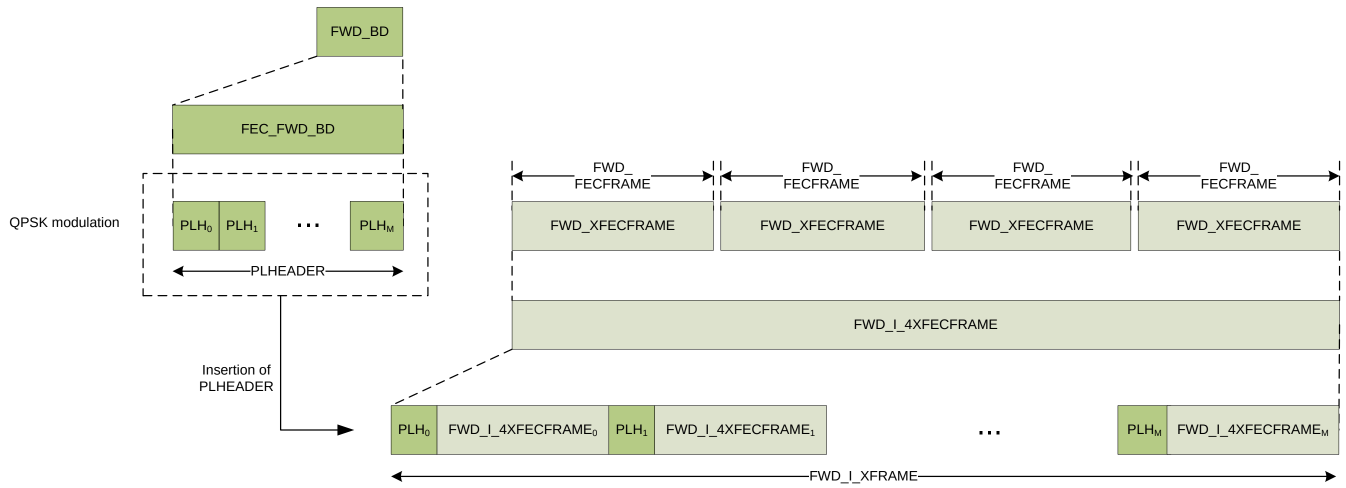
Note: the output PLH_i is a compex symbol ($PLH_i = I_i + jQ_i$).

8.6.2.9.2 Physical Layer Signalling Insertion

D018-COM-FUN-1450

The 64 QPSK PLHEADER symbols shall be uniformly spread through the FWD_I_4XFECFRAME according to the following rule:

1. The FWD_I_4XFECFRAME are divided into 64 fragments of $N_{FWD\_I\_4XFECFRAME\_F}$ = 192 symbols, named FWD_I_4XFECFRAME_F_k (k=0, ..., M; with M=63).
2. At the beginning of each fragment (FWD_I_4XFECFRAME_F_k), one PLHEADER symbol (PLH_k) is inserted starting with the MSB, as represented in the following figure.



The output stream (FWD_I_XFRAME) is $N_{FWD\_I\_XFRAME}$ = 12352-symbols in length.

8.6.2.10 Physical Layer Framing

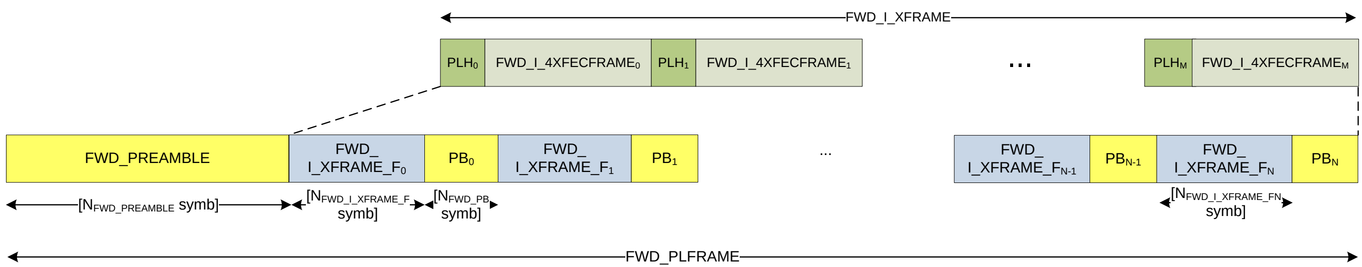
D018-COM-FUN-1460

The Physical Layer Framing module shall perform the following process:

- PB (Pilot Block) insertion.
- FWD_PREAMBLE insertion.

D018-COM-FUN-1470

The input stream of the Physical Layer Framing sub-system shall be a FWD_I_XFRAME and the output a FWD_PLFRAME.



8.6.2.10.1 PB (Pilot Blocks) insertion

D018-COM-FUN-1480

A Pilot Block (PB) shall be composed of N_{FWD_PB} = 24 un-modulated symbols with equal In-phase (I) and Quadrature (Q) components:

$$I_j = Q_j = \frac{\rho}{\sqrt{2}}, \qquad j = 0, \ldots, N_{FWD\_PB} - 1$$

D018-COM-FUN-1490

PB shall be uniformly inserted within the FWD_I_XFRAME according to the following procedure:

1. The FWD_I_XFRAME is divided into fragments called $FWD\_I\_XFRAME\_F_i$ (see D018-COM-FUN-1470), resulting in:
   - 55 fragments of $N_{FWD\_I\_XFRAME\_F}$ = 224 symbols ($FWD\_I\_XFRAME\_F_0$, ..., $FWD\_I\_XFRAME\_F_{54}$)
   - 1 last fragment of $N_{FWD\_I\_XFRAME\_FN}$ = 32 symbols ($FWD\_I\_XFRAME\_F_{55}$)
2. A PB is appended after each FWD_I_XFRAME_Fi, as illustrated in D018-COM-FUN-1470 above.

8.6.2.10.2 Preamble insertion

D018-COM-FUN-1500

The FWD_PREAMBLE shall consist of a $N_{FWD\_PREAMBLE}$ = 100 un-modulated symbols with equal In-phase (I) and Quadrature (Q) components:

$$I_k = Q_k = \frac{\rho}{\sqrt{2}}, \quad k = 0, \ldots, N_{FWD\_PREAMBLE} - 1$$

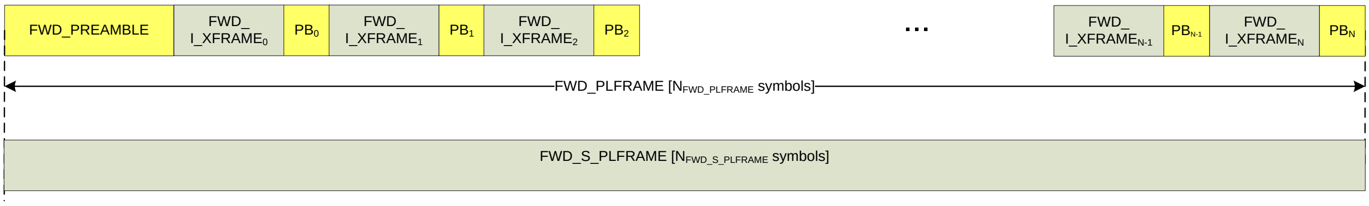
D018-COM-FUN-1510

The FWD_PREAMBLE shall be inserted before the first fragment of FWD_I_XFRAME ($FWD\_I\_XFRAME\_F_0$), as illustrated in D018-COM-FUN-1470 above.

8.6.2.11 Physical Layer Scrambling

D018-COM-FUN-1520

The input stream of the Physical Layer Scrambling module shall be a FWD_PLFRAME and the output stream is a FWD_S_PLFRAME (FWD Scrambled PLFRAME).



Note: the length of the input and output data streams (FWD_PLFRAME and FWD_S_PLFRAME) is the same, as the Physical Layer scrambling does not add redundancy, i.e., $N_{FWD\_PLFRAME}$ symbols = $N_{FWD\_S\_PLFRAME}$ symbols.

D018-COM-FUN-1530

The FWD_PLFRAME shall be randomised for energy dispersal by multiplying the In-phase ($I_{FWD\_PLFRAME}$) and Quadrature ($Q_{FWD\_PLFRAME}$) samples by a complex randomization sequence ($C_{FWD\_I\_SCR}+jC_{FWD\_Q\_SCR}$):

$$I_{FWD\_S\_PLFRAME} = \left(I_{FWD\_PLFRAME} \cdot C_{FWD\_I\_SCR} - Q_{FWD\_PLFRAME} \cdot C_{FWD\_Q\_SCR}\right)$$

$$Q_{FWD\_S\_PLFRAME} = \left(I_{FWD\_PLFRAME} \cdot C_{FWD\_Q\_SCR} + Q_{FWD\_PLFRAME} \cdot C_{FWD\_I\_SCR}\right)$$

The complex randomization sequence is defined in D018-COM-FUN-1550 below.

Note: The whole FWD_PLFRAME is randomized; this includes the FWD_PREAMBLE, the PB and the FWD_I_XFRAME fragments.

D018-COM-FUN-1540

The randomisation sequence ($C_{FWD\_I\_SCR}+jC_{FWD\_Q\_SCR}$) shall be reinitialised at each FWD_PLFRAME.

D018-COM-FUN-1550

The complex scrambling sequence shall be constructed by combining 2 real m-sequences, x and y (generated by means of 2 generator polynomials of degree 18), into a complex sequence, as follows:

1. The sequence x is constructed using the primitive polynomial $h(x) = 1 + x^7 + x^{18}$
2. The sequence y is constructed using the primitive polynomial $g(y) = 1 + y^5 + y^7 + y^{10} + y^{18}$
3. The sequence depending on the chosen scrambling code number n is denoted z_n in the sequel and x(i), y(i) and $z_n(i)$ denote the i^{th} symbol of the sequence x, y and z_n respectively. Then, the m-sequences x and y are constructed as:
   a. Initial conditions:
      - x is constructed with $x(0) = 1, x(1) = x(2) = \ldots = x(16) = x(17) = 0$
      - y is constructed with $y(0) = y(1) = \ldots = y(16) = y(17) = 1$

b. Recursive definition of subsequent symbols
   - $x(i+18) = [x(i+7) + x(i)]mod2, \quad i = 0, \dots, 2^{18} - 20$
   - $y(i+18) = [y(i+10) + y(i+7) + y(i+5) + y(i)]mod2, \quad i = 0, \dots, 2^{18} - 20$

c. The nth Gold code sequence zn, n=0,1,2, ..., 218-2, is defined as

$$z_n(i) = \left[x\left((i+n)mod(2^{18}-1)\right) + y(i)\right]mod(2), i = 0, \dots, 2^{18} - 2$$

d. These binary sequences are converted to integer valued sequences Rn (Rn assuming values 0, 1, 2 and 3) by the following transformation:

$$R_n(i) = 2 \cdot z_n\left((i+131072)mod(2^{18}-1)\right) + z_n(i), \quad i = 0, 1, \dots, 13795$$

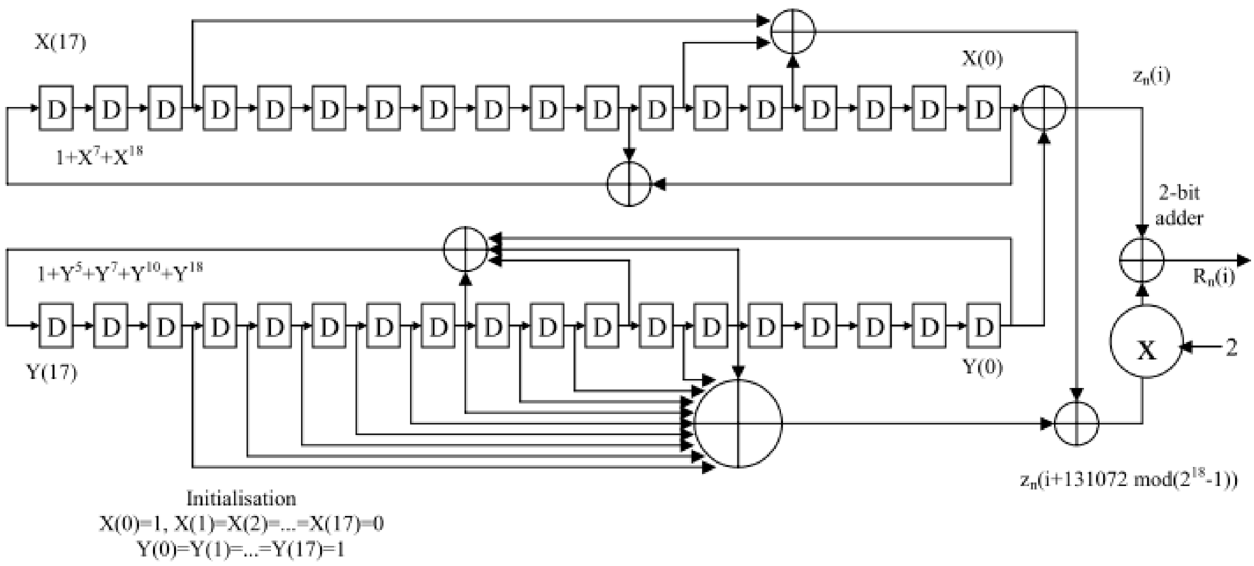
e. Finally, the nth complex scrambling sequence CI(i)+jCQ(i) is defined as:

$$C_{FWD\_I\_SCR}(i) + jC_{FWD\_Q\_SCR}(i) = exp\left(jR_n(i) \cdot \frac{\pi}{2}\right)$$

R_n	$exp\left(jR_n(i) \cdot \frac{\pi}{2}\right)$	$I_{FWD\_S\_PLFRAME}$	$Q_{FWD\_S\_PLFRAME}$
0	1	I	Q
1	j	-Q	I
2	-1	-I	-Q
3	-j	Q	-I

D018-COM-FUN-1551

The Physical Layer complex scrambling sequence shall be generated with n=0. The following figure shows the block diagram for pseudo-randomization sequence generation for n=0.



8.6.2.12 Base-band pulse shaping and quadrature modulation

D018-COM-FUN-1560

After the physical layer scrambling, the FWD_S_PLFRAME shall be shaped and quadrature modulated.

D018-COM-FUN-1570

I and Q signals shall be shaped using a Square Root Raised Cosine with a roll-off factor α=0.2.

D018-COM-FUN-1580

The base-band SRRC filter shall have a theoretical function defined by the following expression.

$$H(f) = 1 \qquad \text{for } |f| < f_N(1-\alpha)$$

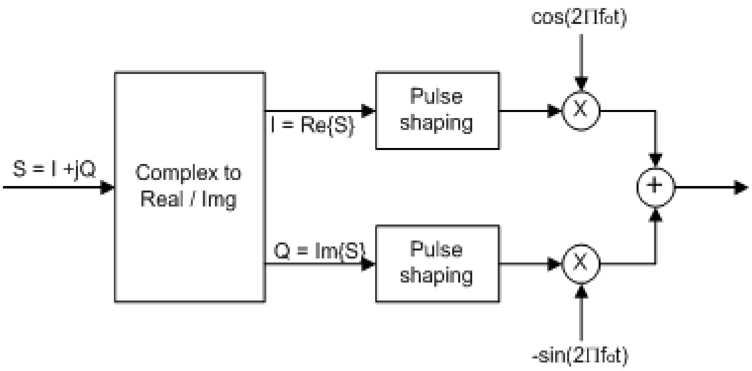
$$H(f) = \left\{ \frac{1}{2} + \frac{1}{2} \sin \frac{\pi}{2 f_N} \left[\frac{f_N - |f|}{\alpha} \right] \right\}^{1/2} \qquad \text{for } f_N(1-\alpha)$$

$$H(f) = 0 \ \text{ for } |f| > f_N(1+\alpha),$$

where $f_N = R_s/2$ is the Nyquist frequency and α is the roll-off factor.

D018-COM-FUN-1590

The quadrature modulation shall be performed by multiplying the In-phase component (I) by $\cos(2\pi f_0 t)$ and the Quadrature component (Q) by $-\sin(2\pi f_0 t)$ and by adding both signals (I and Q), as represented in the following figure.



8.7 User plane return link specification

8.7.1 Link layer specification

D018-COM-FUN-1601

For the RTN link ARQ and encapsulation processes, the mask 0xFC shall be applied to the GES ID field.

Note: only the first six most significant bits of the GES ID field will be used. This requirement has been included for redundancy support purposes. Refer also to D018-COM-FUN-0469 and D018-COM-FUN-0715 above.

8.7.1.1 Random access and scheduling

D018-COM-FUN-1605

The UT shall perform the congestion control procedure specified by D018-COM-FUN-1610 to the following entities:

- NPDUs at the moment they are received from upper layers.
- The following signalling messages, at the moment they are received from control plane:
  - ARQ-ACKs signalling messages.
- The set of LPDUs (or ARQ block) selected for retransmission at the moment their retransmission timer triggers.

D018-COM-FUN-1610

The UT shall apply the congestion control mechanism described below to the traffic indicated in D018-COM-FUN-1605 above:

1. The variable '*npdu_ntx*' is bound to the NPDU or signalling message when it is received from upper layer or control plane, respectively. It is initialized to 1.
2. A random delay, '*backoff_delay*', is computed as indicated below:

$$backoff\_delay = \begin{cases} \sum_{k=1}^{K} delay\,[k], & lcc\_tx\_backoff > 0 \\ 0, & lcc\_tx\_backoff = 0 \end{cases}$$

   where '*delay*' is a random number that follows an exponential distribution whose mean is:

$$1/\lambda = lcc\_tx\_backoff \cdot 2^{(npdu\_ntx-1)}$$

   '*K*' is a realization of a random variable following a shifted geometric distribution with probability equal to '*lcc_persistence*'. The CDF of the shifted geometric distribution is:

$$CDF = 1 - (1 - lcc\_persistence)^k, with\ support\ k \in \{1, 2, 3, ...\}$$

3. The variable '*npdu_ntx*' for this NPDU is incremented by one.
4. A delay of '*backoff_delay*' ms is applied to the NPDU, ARQ block or signalling message before forwarding it to the transmission scheduler.

The values of '*lcc_tx_backoff*' and '*lcc_persistence'* are resolved from congestion control parameters '*tx_backoff*' and '*persistence*' distributed by the GS in the *CC_Config* signalling structure.

The ARQ-ACKs use the congestion control parameters '*tx_backoff*' and '*persistence*' associated with the '*CC_category_id*' with value '*High*'.

D018-COM-FUN-1615

The UT transmission scheduler shall satisfy the constraints and targets stated below when the QoS information specified in D018-COM-FUN-0580 is available:

- Voice traffic and signalling messages follow a scheduling policy with:
  - Voice having strict priority over signalling messages.
  - Signalling messages having strict priority over data traffic.
- For data traffic, the UT schedules must satisfy the constraints imposed by the TD95 and ET QoS parameters.

D018-COM-FUN-1625

In the event that the QoS information specified in D018-COM-FUN-0580 is available, the UT shall be able to compute the *expiration_timeout* of NPDUs as indicated below:

expiration_timeout = ET - rl_path_and_processing_time

The *expiration_timeout* indicates the time allowed for the NPDU transmission before the ET QoS constraint is violated.

The parameter *rl_path_and_processing_time* is distributed by the GS through signalling messages to inform the UTs about the estimated latency introduced by propagation delay and GS processing time.

D018-COM-FUN-1640

The UT shall de-queue and drop the NPDUs, including all its associated LPDUs, that are known to fail to satisfy the ET QoS constraint because they could not be transmitted within the time interval indicated by *expiration_timeout*.

D018-COM-FUN-1650

The ARQ field of the RL Encapsulation header, as defined in D018-COM-FUN-1780 below, along with the UT's L2 address shall be used for the in-order reassembly of the received NPDU's fragments.

8.7.1.2 ARQ protocol

8.7.1.2.1 ARQ header and ACK format

D018-COM-FUN-1660

The ARQ header format shall include the following fields:

Flow ID	Packet Count	R	P/F	Fragment Count
4 bits	4 bits	1 bit	1 bit	6 bits

The fields have the same meaning as in D018-COM-FUN-0720 above. Reserved bit R is set to zero.

Note: The only difference with D018-COM-FUN-0720 *is that in the event of fragmentation the 3 bit FID field defined in the RTN link encapsulation header is used (see D018-COM-FUN-1780). So, for this case, the remaining most significant bit of the Flow ID field is set to zero. In any case the encapsulation ARQ field is 16 bits long due to byte alignment.*

D018-COM-FUN-1670

The ARQ ACK format is as specified in D018-COM-FUN-0730 above.

8.7.1.2.2 ARQ procedure

This section provides the specification of the return link ARQ procedure.

The following definitions are used to support the description of the procedure:

- *ARQ Block* as defined for the forward link ARQ procedure.
- T^{Tx} as defined for the forward link ARQ procedure.
- *arq_retx_timeout* as defined for the forward link ARQ procedure. In the case of the RL, the GS distributes the possible values of *arq_retx_timeout* in the CC_config signalling structure.

D018-COM-FUN-1690

The same ARQ procedure specified in 8.6.1.2 shall be applied in the return link with the particularities listed below:

- The ARQ procedure uses the identifier that results after masking the GSE ID with the bit pattern specified in D018-COM-FUN-1601.
- If there is a retransmission event, when the *ReTx_Timer* triggers, the UT selects all the LPDUs of the *active_block* for retransmission by setting '*nReTx*' to the number of LPDUs in the *active_block*.
- The LPDUs selected for retransmission are forwarded to the Congestion Control procedure specified in D018-COM-FUN-1610.

D018-COM-FUN-1720

The UT shall select the value of *arq_retx_timeout* from the set of values of the parameter *retransmission_timeout* distributed by the GS in the *CC_config* signalling structure.

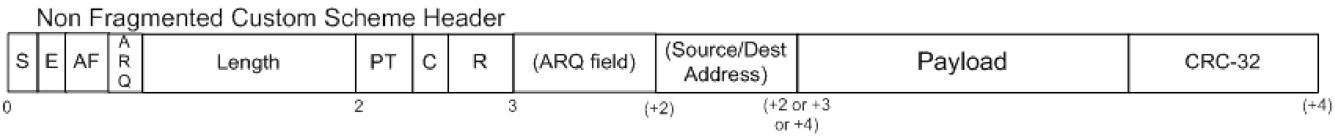
8.7.1.3 Encapsulation

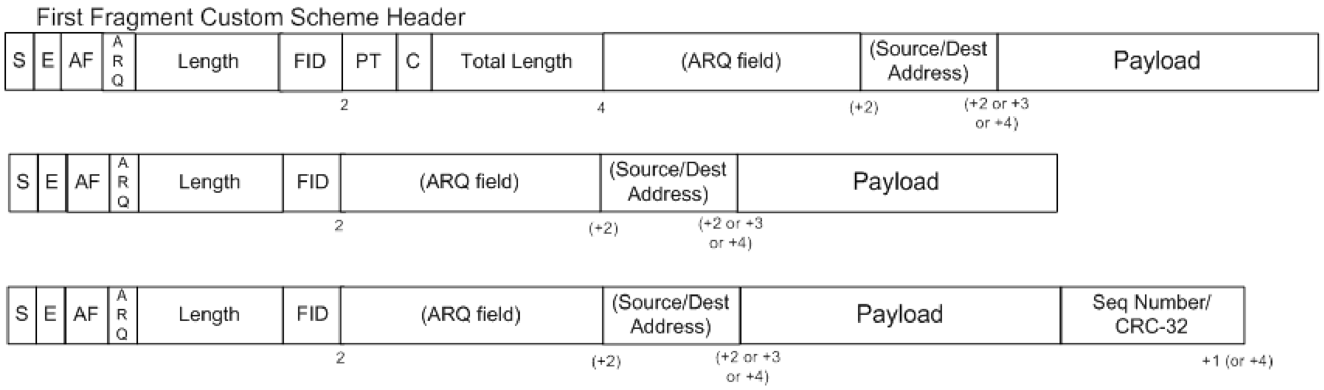
D018-COM-FUN-1770

The return link encapsulation scheme requires that a PPDU payload always starts with the encapsulation header of the first LPDU. Therefore, LPDU fragmentation between PPDU payloads is not allowed.

D018-COM-FUN-1780

The return link encapsulation scheme used shall comply with the following.





where

Field	Length	Description
S and **E** flags: Start and End bits	1 bit each	These fields indicate if the packet is non-fragmented (11), of if it is the first fragment (10), the last fragment (01) or an intermediate fragment (00) of a fragmented packet. S and E bits set to 0 also permits signal padding when Length value field (see below) is set to 0 as well, meaning that the rest of the physical payload is composed of padding bits (0 bits). This padding signalling is required only when the physical payload remaining space after the last inserted fragment is longer or equal to 4 bytes; otherwise the rest of the L1 payload is composed of padding bits by default.
AF field: Address Format	2 bits	This field indicates the address format that will be contained in the encapsulation header. It can take the following values: **00**: No address field/address format re-use (re-use the address field of the previous LPDU in the same L1 payload) **01**: 2 byte address field: UT source address **10**: 3 byte address field: 2 byte source UT address + 1 byte destination GS element address **11**: 4 byte address field: 3 byte source UT ICAO address + 1 byte destination GS element address
ARQ flag	1 bits	This field indicates whether the ARQ process is used. If set to 1 it implies an additional 2 byte field used for ARQ.
Length field	11 bits not fragmented/ 8 bits fragmented	This field indicates the length of the L2 payload size for the current fragment (for fragmented packet) or the total length of the payload (for non-fragmented packet). This length enables physical payload sizes up to 259 bytes (this length is sufficient in regards to the considered physical burst size over the return link, which is below 259 bytes).
PT field: Payload Type	3 bits	This field indicates the type of the contained payload. PT field can take the following values: **000**: This value of the PT field indicates that an ATN/IPS NPDU is encapsulated in the current LPDU (without header compression). **001**: This value of the PT field indicates that an ATN/OSI NPDU is encapsulated in the current LPDU. The ATN/OSI protocol self identifies whether the packet is compressed or not. **010**: This value of the PT field indicates that signalling information is encapsulated in the current LPDU. **011**: This value of the PT field indicates that an ATN/IPS header compressed NPDU is encapsulated in the current LPDU. The other values are reserved for future use.

Field	Length	Description
C flag: CRC Presence	1 bits	When the value of the flag is set to "1", this field indicates a 4 byte CRC field is present as a trailer in the current packet when un-fragmented or in the last fragment, when fragmented. When fragmented, if the C flag is set to "0", a 1-byte sequence number is added to the last fragment and is used by the re-assembly process to detect fragment losses.
R field: Reserved	4 bits	This field is reserved for future use.
ARQ field: ARQ Information	2 bytes	This optional field contains the information required for the ARQ process when the ARQ bit is set to 1. It shall comply with D018-COM-FUN-1660. When fragmented, the ARQ mechanism re-uses the Flow ID field already present in the encapsulation header. When non-fragmented, the three fields of the ARQ are required (FID, PC and FC).
Source/Dest Address field	optional field of 2, 3 or 4 bytes	This field is optional and contains the source and destination addresses as indicated by the Address Format (AF) field.
FID field: Flow ID	3 bits	This information is used by the ARQ process and/or by the re-assembly process for fragmented LSDUs. It shall comply with the Flow Identifier field of the ARQ header defined in D018-COM-FUN-1660. The presence of this field is mandatory for LPDUs associated with fragmented LSDUs.
Total Length field	12 bits	This field indicates the total length of the L3 payload when fragmented. This length enables NPDU sizes up to 4095 bytes.
Seq Number/CRC-32 field	1 or 4 bytes	When indicated by the C bit (C flat set to value "1"), a trailer is added at the end of the last fragment when fragmented or after the payload when un-fragmented with a 4-byte CRC. The CRC-32 is computed as indicated in D018-COM-FUN-1070 and D018-COM-FUN-1080, considering the PNPDU as input stream. When fragmented, if the C flag is set to 0, a 1-byte sequence number is added and used by re-assembly process to detect fragment loss.This number is a cyclic counter of fragmented packets (per Flow ID).

8.7.1.4 Security

Refer to 8.3.

8.7.2 Physical layer specification

8.7.2.1 Burst types

D018-COM-FUN-1790

On the return link the following burst types shall be implemented:

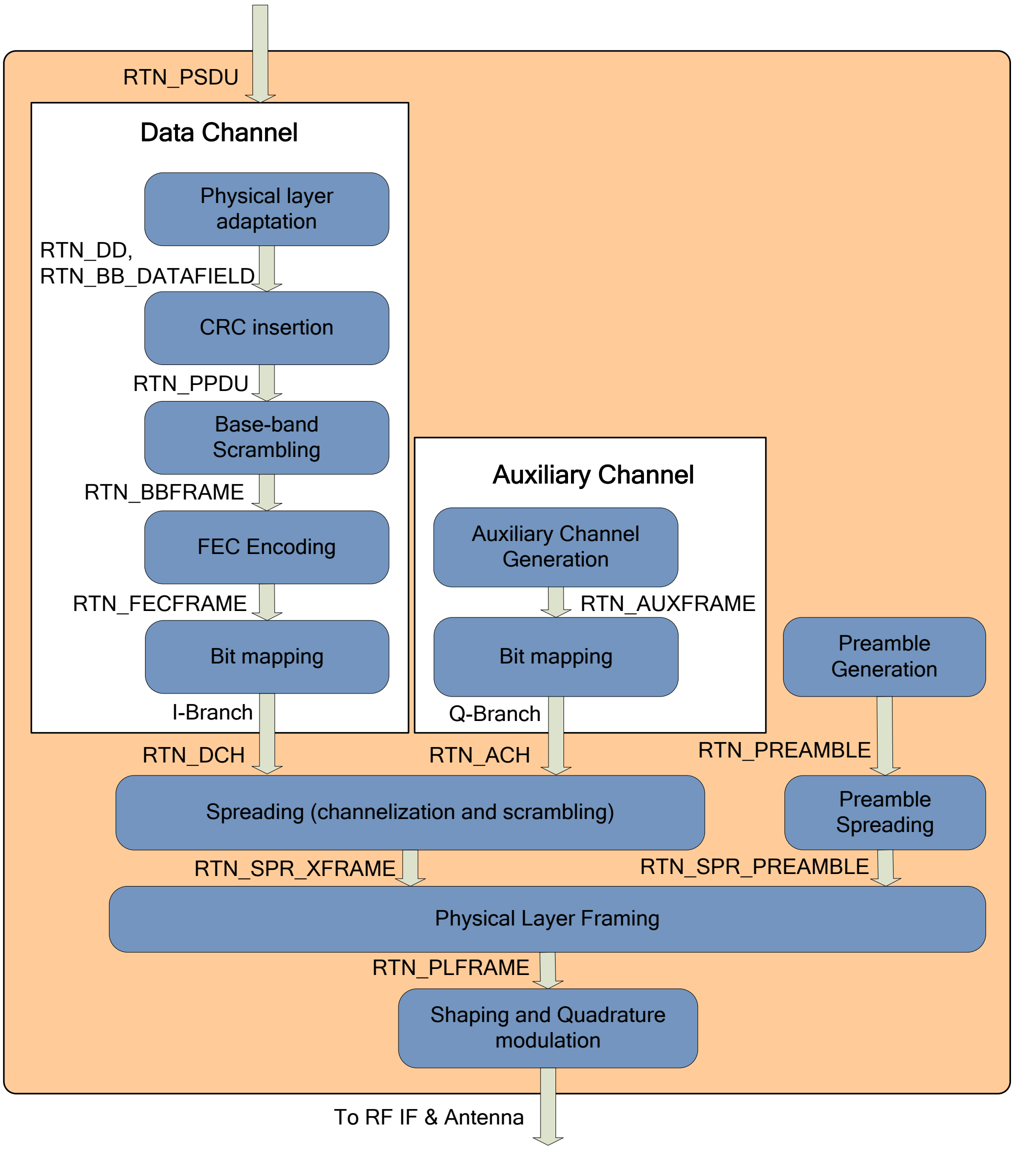
- RACH burst (Random Access Channel)

8.7.2.2 Burst waveform generator

D018-COM-FUN-1800

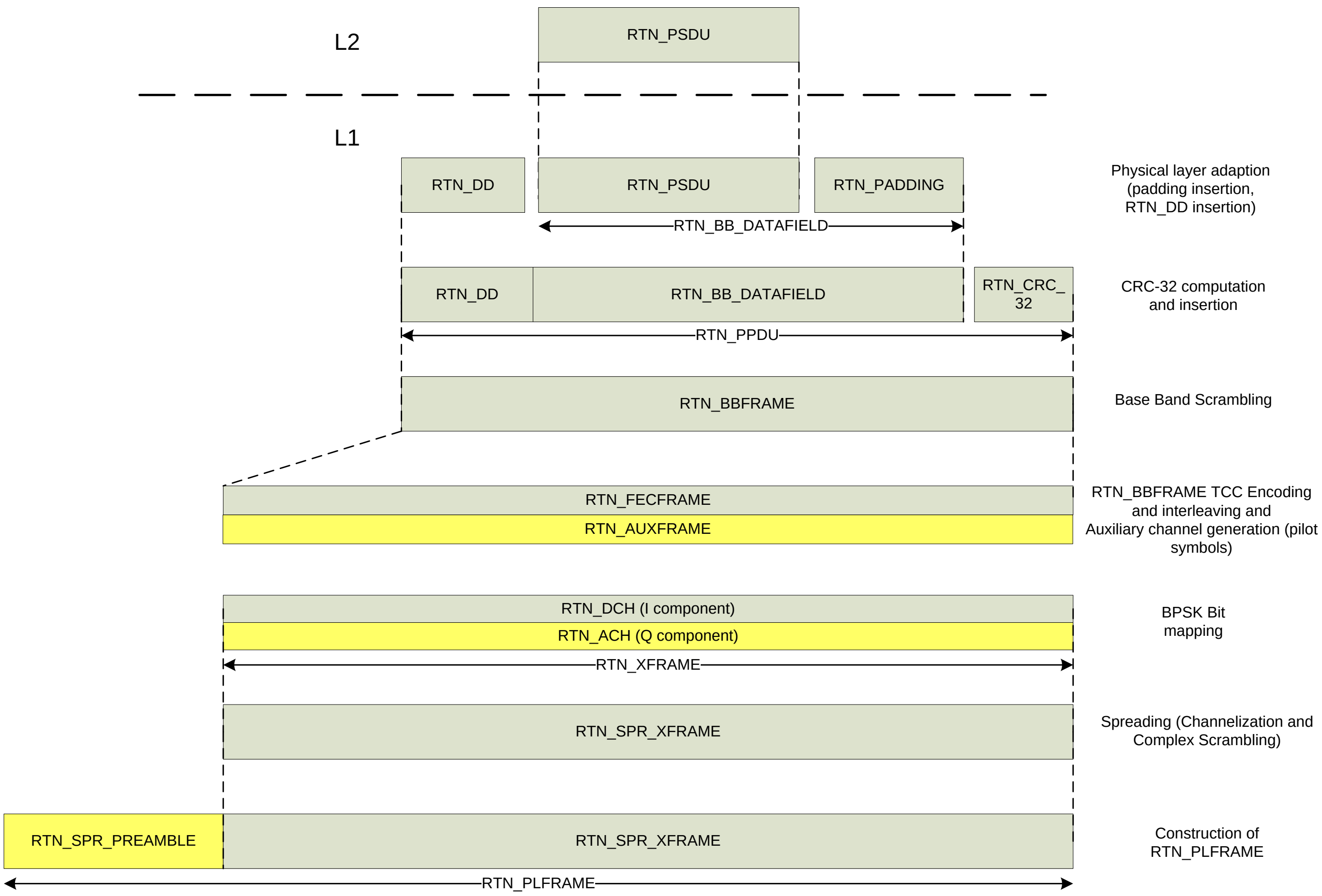
The burst waveform generation shall be applied to the RTN_PSDU and composed of a sequence of functional modules as represented in the figure below. The functional modules are:

- Physical Layer Adaptation
- CRC Insertion
- Base-band Scrambling
- FEC Encoding, which includes the inner encoding and the bit interleaver
- Auxiliary channel generation
- Bit Mapping into Constellation
- Spreading
- Preamble generation
- Preamble spreading
- Physical Layer Framing
- Base-band Pulse Shaping and Quadrature Modulation



D018-COM-FUN-1810

The RTN Link bursts shall follow the frame format presented in the following figure.



D018-COM-FUN-1820

The RACH burst shall support the following configurations.

RACH Configuration ID	Chip rate (kchip/s)	SF length	Mod	Code rate	Data Word Size (bits) - RTN_PPDU
RACH_CR$_{160}$_SF$_{16}$_DB$_{512}$	160	16	BPSK	1/3	512
RACH_CR$_{160}$_SF$_{4}$_DB$_{2048}$	160	4	BPSK	1/3	2048
RACH_CR$_{160}$_SF$_{16}$_DB$_{288}$	160	16	BPSK	1/3	288
RACH_CR$_{160}$_SF$_{4}$_DB$_{976}$	160	4	BPSK	1/3	976

Note: these configurations correspond to the Data Channel.

8.7.2.3 Physical Layer Adaptation

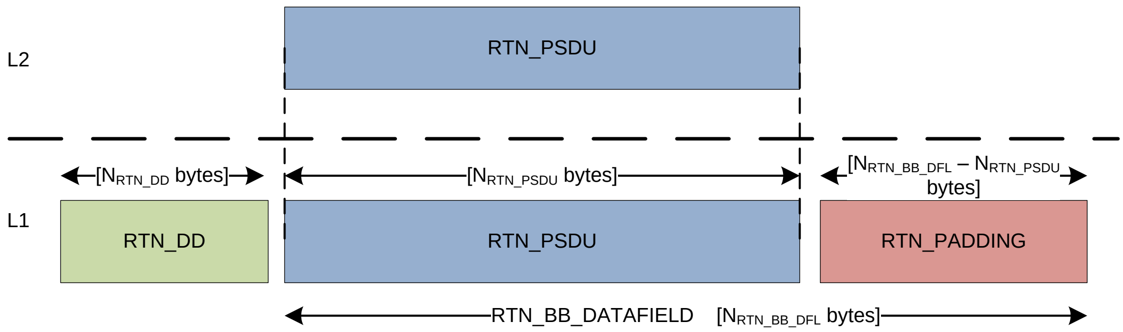
D018-COM-FUN-1830

The Physical Layer Adaptation module shall perform:

- Interface with Layer 2
- Padding insertion
- RTN_DD (RTN Data Descriptor) insertion

D018-COM-FUN-1840

The input stream of the Physical Layer Adaptation module shall be a RTN_PSDU and the output stream a RTN_DD Header followed by a RTN_BB_DATAFIELD, as detailed in the following figure.



D018-COM-FUN-1850

A single RTN PSDU shall be mapped onto a single burst.

8.7.2.3.1 Interface with Layer 2

D018-COM-ITF-1860

The Physical Layer Adaptation module shall map RTN_PSDU of up to 4 types of layer 2 entities (or protocols):

- RL Encapsulation, as defined in D018-COM-FUN-1780 above
- Protocol 2 (RFU)
- Protocol 3 (RFU)
- Protocol 4 (RFU)

D018-COM-ITF-1870

The RTN_PSDU size (in bytes) shall be variable in the range depending on the selected RACH configuration. The maximum RTN_PSDU size (Max. N_{RTN_PSDU}) are as follows:

RACH Configuration ID	Max. N_{RTN_PSDU} [Maximum RTN_PSDU size (bytes)]
$RACH\_CR_{160}\_SF_{16}\_DB_{512}$	58 bytes
$RACH\_CR_{160}\_SF_{4}\_DB_{2048}$	250 bytes
$RACH\_CR_{160}\_SF_{16}\_DB_{288}$	30 bytes
$RACH\_CR_{160}\_SF_{4}\_DB_{976}$	116 bytes

Note: Maximum RTN PSDU size (Max. N_{RTN_PSDU}) corresponds with the RTN_BB_DATAFIELD size ($N_{RTN\_BB\_DFL}$).

8.7.2.3.2 Padding insertion

D018-COM-FUN-1880

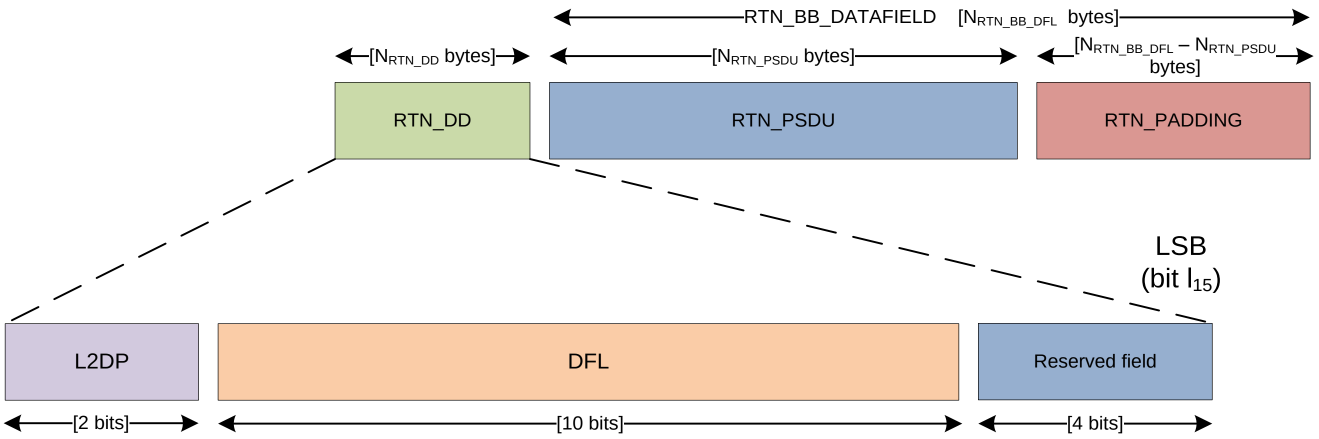
($N_{RTN\_BB\_DFL}$ - N_{RTN_PSDU}) bytes shall be appended after the RTN_PSDU according to the figure in D018-COM-FUN-1890 below. The content of the padding bytes is "0x00".

8.7.2.3.3 RTN_DD (RTN Data Descriptor) insertion

D018-COM-FUN-1890

A fixed length header (RTN_DD) of N_{RTN_DD}= 2 bytes (bits I_0-I_{15}) shall be inserted in front of the RTN_BB_DATAFIELD according to the figure below. The RTN_DD header contains the following fields.

Bits	Field	Field size	Description
I_0-I_1	L2DP	2 bits	It indicates the L2 protocol used in the RTN_PSDU
I_2-I_{11}	DFL	10 bits	It contains the RTN_PSDU size in bytes. Range [0-1023].
I_{12}-I_{15}	Reserved field	4 bits	Reserved for future use



D018-COM-FUN-1900

The L2DP field shall indicate which L2 entity is the client of the burst. It is a 2-bit field coded as follows.

I_0 I_1 (L2DP)	Description
00	RL Encapsulation
01	RFU
10	RFU
11	RFU

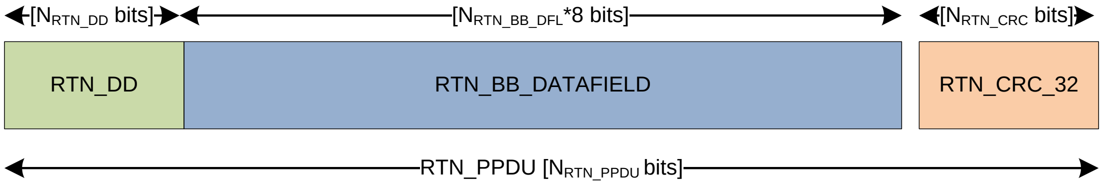
8.7.2.4 CRC insertion

D018-COM-FUN-1910

The RTN CRC Insertion module shall compute the CRC parity bits in order to detect erroneous packets at the receiver end and to provide a packet quality indicator.

D018-COM-FUN-1920

The input stream of the RTN CRC Insertion module shall be a RTN_DD header followed by a RTN_BB_DATAFIELD and the output stream a RTN_PPDU, as illustrated in the following figure.



D018-COM-FUN-1930

The RTN_CRC_32 shall be computed over:

- the RTN_DD header
- the RTN_BB_DATAFIELD

by a systematic 32-bit CRC encoder and appended after the RTN_BB_DATAFIELD, as illustrated in D018-COM-FUN-1920 above.

D018-COM-FUN-1940

The content of the RTN_CRC_32 field shall be computed as described in D018-COM-FUN-1070 and D018-COM-FUN-1080 above.

8.7.2.5 Base-band scrambling

D018-COM-FUN-1950

The Base-band scrambling module shall add a binary pseudo-noise sequence to the input data stream in order to randomise the binary transitions in the output stream.

D018-COM-FUN-1960

The input stream of the Base-band scrambling module shall be a RTN_PPDU and the output stream a RTN_BBFRAME.

Note: the length of the input and output data streams (RTN_PPDU and RTN_BBFRAME) is the same, as the Base-band scrambling does not add redundancy.

D018-COM-FUN-1970

The complete RTN_PPDU shall be randomized using the Base-band scrambling. The binary pseudo-noise sequence is synchronous with the RTN_PPDU, starting with the MSB of the RTN_PPDU.

D018-COM-FUN-1980

The binary pseudo-random sequence (scrambling sequence) shall be generated by a linear shift register with connection polynomial as follows:

$$h(D) = 1 + D^{14} + D^{15}$$

D018-COM-FUN-1990

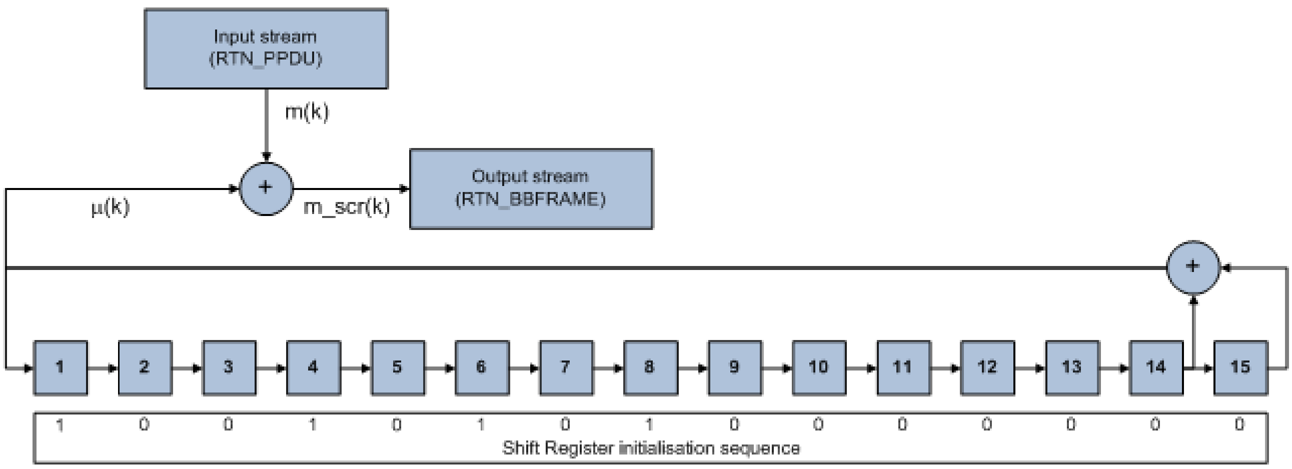
The RTB_BBFRAME shall be generated by adding in modulo 2 the pseudo-random sequence with the connection polynomial specified in D018-COM-FUN-1980 to the input data stream (RTN_PPDU), as shown in the figure below. The procedure to generate the RTN_BBFRAME from a RTN_PPDU is as follows:

1. At the start of every RTN_PPDU, the contents of the shift register cells are initialised with the following sequence (1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0), as indicated in the Base-band scrambler block diagram below.
2. The first value of the pseudo-noise sequence is μ(0) = 0, being the modulo-2 sum of the two last elements in the shift register after initialization. For each input bit of the RTN_PPDU frame, the shift register is clocked once to generate the next element of the pseudo-noise sequence.
3. After each clocking of the shift register, the output bit of the RTN_BBFRAME frame is computed according to the rule:

   $$m\_scr(k) = [m(k) + \mu(k)]mod2$$

   where k represents the kth clocking of the shift register.

The shift register is clocked a number of times equal to the number of inputs bits of RTN_PPDU frame (N_{RTN_PPDU}). The MSB of the RTN_PPDU is randomised first.



8.7.2.6 FEC encoding

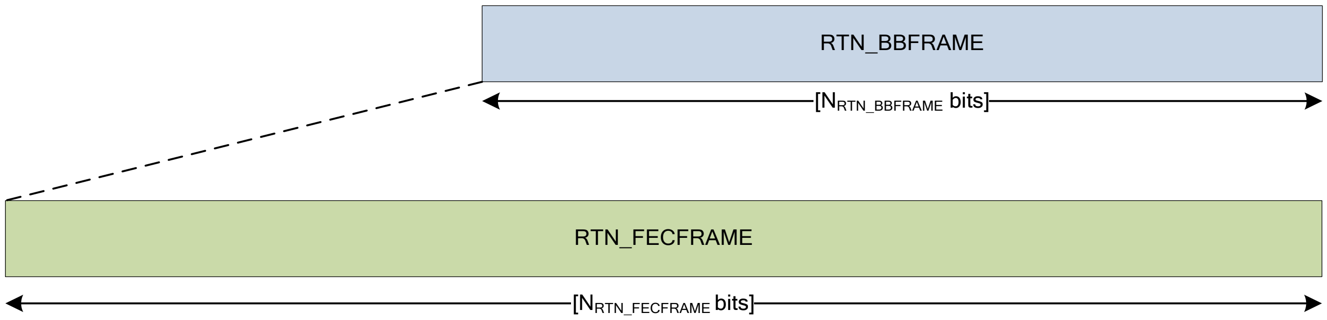
D018-COM-FUN-2000

The FEC Encoding module shall perform the following functions:

- Inner channel coding (TCC)
- Bit interleaving

D018-COM-FUN-2010

The input stream of the FEC encoding module shall be a RTN_BBFRAME and the output stream a RTN_FECFRAME, as illustrated in the following figure.



D018-COM-FUN-2020

The RTN_BBFRAME shall be encoded with 1/3 code rate using the inner coding described in section 8.7.2.6.1.

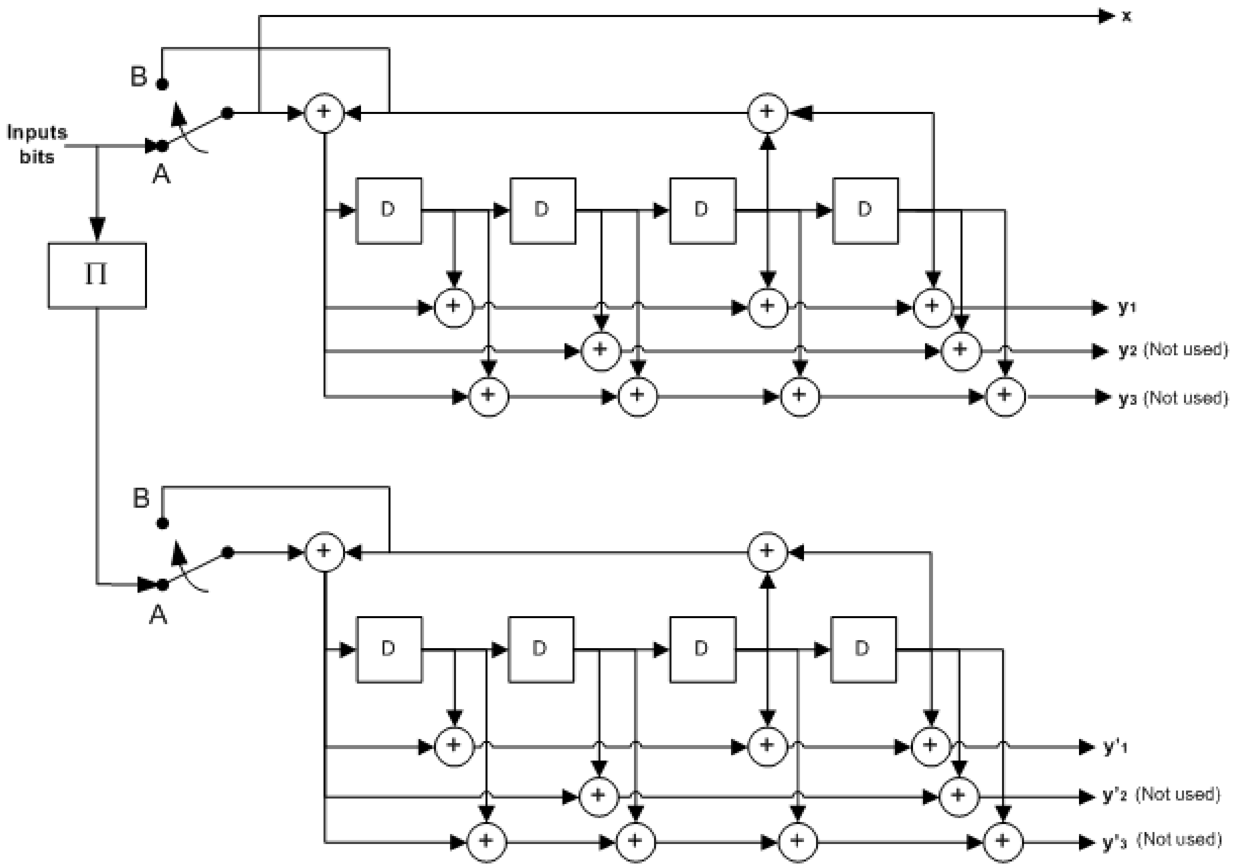
8.7.2.6.1 Inner encoding (TCC)

D018-COM-FUN-2030

The Forward Error correction scheme on the return link shall be Turbo Convolutional Codes.

D018-COM-FUN-2040

The Turbo code architecture shall be a Parallel Concatenated Convolutional Code (PCCC) with two identical 16-state binary, recursive and systematic convolutional encoders and a Turbo code internal interleaver, as depicted in the following figure.



(Note1: y_2, y_3, y'_2, and y'_3 are not used for the code rate 1/3.)

(Note2: Π represents the TCC internal interleaver which is specified in requirement D018-COM-FUN-2060.)

The Turbo encoder output is computed according to the following procedure (code rate 1/3):

1. Both 16-state constituent encoders are initialised with 0 in all the shift registers before coding a block (initial state)
2. All switches are set in the A position
3. The encoded data is generated by clocking the constituent encoders $N_{RTN\_BBFRAME}$ times. MSB of RTN_BBFRAME is inserted first in the TCC encoder. The constituent encoder output for each bit period k (k< $N_{RTN\_BBFRAME}$) is the output sequence (k represents the k^{th} clocking of the shift register):

$$x(k),\ y_1(k),\ y'_1(k)$$

4. Once the last input bit has been inserted in the constituent encoders (LSB of RTN_BBFRAME), the switches are set to position B
5. Then the tail (trellis termination) bits are generated by clocking 4 times the constituent encoders. The tail bits are appended after the encoded information block. The tail bits are transmitted in the same order as in step 3 and are:

$$x(N_{RTN\_BBFRAME}),\ y1(N_{RTN\_BBFRAME}),\ y'_1(N_{RTN\_BBFRAME}),\ \ldots,\ x(N_{RTN\_BBFRAME+3}),\ y_1(N_{RTN\_BBFRAME+3}),\ y'_1(N_{RTN\_BBFRAME+3})$$

6. The transmitted sequence is:

$x(0)$, $y_1(0)$, $y'_1(0)$, $x(1)$, $y_1(1)$, $y'_1(1)$, ..., $x(N_{RTN\_BBFRAME-1})$, $y_1(N_{RTN\_BBFRAME-1})$, $y'_1(N_{RTN\_BBFRAME-1})$, $x(N_{RTN\_BBFRAME})$, $y_1(N_{RTN\_BBFRAME})$, $y'_1(N_{RTN\_BBFRAME})$, ..., $x(N_{RTN\_BBFRAME+3})$, $y_1(N_{RTN\_BBFRAME+3})$, $y'_1(N_{RTN\_BBFRAME+3})$

D018-COM-FUN-2050

The transfer function of the 16-state constituent code shall be:

$$G(D) = \left[1 \quad \frac{n_0(D)}{d(D)} \quad \frac{n_1(D)}{d(D)} \quad \frac{n_2(D)}{d(D)}\right]$$

where

$$d(D) = 1 + D^3 + D^4$$

$$n_0(D) = 1 + D + D^3 + D^4$$

$$n_1(D) = 1 + D^2 + D^4$$

$$n_2(D) = 1 + D + D^2 + D^3 + D^4$$

(Note: for code rate 1/3 only d(D) and n_0(D) are required.)

D018-COM-FUN-2060

The TCC internal interleaver, represented by the symbol Π in requirement D018-COM-FUN-2040 above, shall interleave the RTN_BBFRAME according to the sequence of interleaver output addresses equivalent to those generated by the procedure illustrated in the following figure and described below.

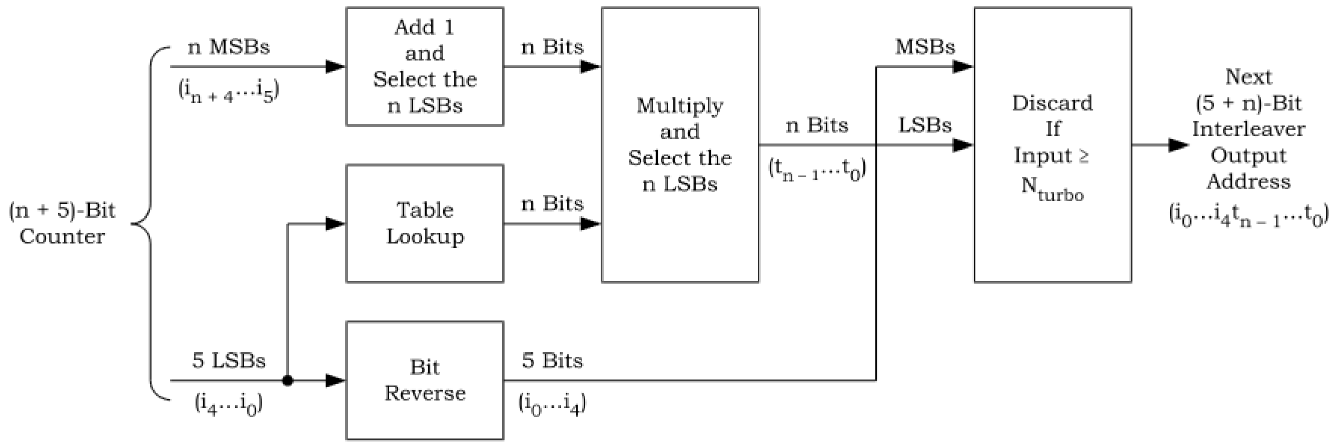
The TCC internal interleaver functionality is such that the input bits (RTN_BBFRAME) are readdressed following a certain mapping sequence. This mapping sequence is generated following the process defined in the 3GPP2 standard and described below.

1. Determine the turbo interleaver parameter *n*, where *n* is the smallest integer such that $N_{RTN\_BBFRAME} \le 2^{n+5}$.
2. Initialise an *(n+5)-bit* counter to *0.*

   (Note: the (n+5)-bit counter is represented as i_{n+4}, i_{n+3}, ..., i_5, ..., i_1, i_0.)
3. Extract the *n* most significant bits (MSBs) from the counter (i_{n+4}, i_{n+3}, ..., i_5) and add one to generate a new value. Then, discard all bits except the *n* least significant bits (LSBs) of this value.
4. Obtain the *n-bit* output of the lookup table defined below with a read address equal to the five LSBs of the counter (i_4, i_3, i_2, i_1, i_0). Note that this table depends on the value of *n*.
5. Multiply the values obtained in Steps 3 and 4 and discard all bits except the *n* LSBs (t_{n-1}, ..., t_0).
6. Bit-reverse the five LSBs of the counter (i_0, ..., i_4).

7. Generate an output address that has its MSBs equal to the value obtained in Step 6 and its LSBs equal to the value obtained in Step 5 (i_0, ..., i_4, t_{n-1}, ..., t_0).
8. Accept the output address if it is less than $N_{RTN\_BBFRAME}$, otherwise discard it.
9. Increment the counter and repeat from Step 3 to Step 8 until all $N_{RTN\_BBFRAME}$ interleaver output addresses are obtained.

Note: the valid range of input and output addresses is from 0 to $N_{RTN\_BBFRAME} - 1$.



The lookup table included in the algorithm is depicted in the following table.

Table index (i_4, ..., i_0)	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10
0	1	5	27	3	15	3	13	1
1	1	15	3	27	127	1	335	349
2	3	5	1	15	89	2	87	303
3	5	15	15	13	1	83	15	721
4	1	1	13	29	31	19	15	973
5	5	9	17	5	15	179	1	703
6	1	9	23	1	61	19	333	761
7	5	15	13	31	47	99	11	327
8	3	13	9	3	127	23	13	453
9	5	15	3	9	17	1	1	95
10	3	7	15	15	119	3	121	241
11	5	11	3	31	15	13	155	187
12	3	15	13	17	57	13	1	497
13	5	3	1	5	123	3	175	909
14	5	15	13	39	95	17	421	769
15	1	5	29	1	5	1	5	349

16	3	13	21	19	85	63	509	71
17	5	15	19	27	17	131	215	557
18	3	9	1	15	55	17	47	197
19	5	3	3	13	57	131	425	499
20	3	1	29	45	15	211	295	409
21	5	3	17	5	41	173	229	259
22	5	15	25	33	93	231	427	335
23	5	1	29	15	87	171	83	253
24	1	13	9	13	63	23	409	677
25	5	1	13	9	15	147	387	717
26	1	9	23	15	13	243	193	313
27	5	15	13	31	15	213	57	757
28	3	11	13	17	81	189	501	189
29	5	3	1	5	57	51	313	15
30	5	15	13	15	31	15	489	75
31	3	5	13	33	69	67	391	163

D018-COM-FUN-2070

The TCC coding shall support the following RACH block sizes reported in the following table.

RACH Configuration ID	Code rate	RTN_BBFRAME (bits)	Tail bits	RTN_FECFRAME (bits)
$RACH\_CR_{160}\_SF_{16}\_DB_{512}$	1/3	512 bits	12 bits	1548 bits
$RACH\_CR_{160}\_SF_{4}\_DB_{2048}$	1/3	2048 bits	12 bits	6156 bits
$RACH\_CR_{160}\_SF_{16}\_DB_{288}$	1/3	288 bits	12 bits	876 bits
$RACH\_CR_{160}\_SF_{4}\_DB_{976}$	1/3	976 bits	12 bits	2940 bits

8.7.2.6.2 Bit interleaving

D018-COM-FUN-2080

The output of the TCC encoder shall be bit interleaved using a block interleaver (row/column) with the parameters defined in the following table.

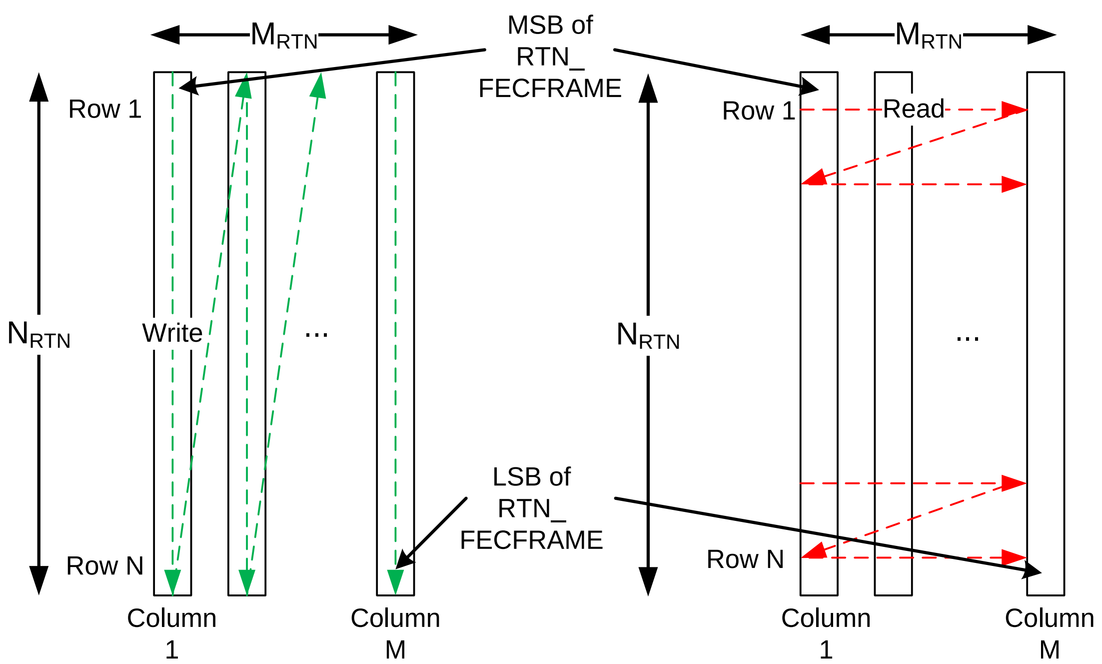
RACH Configuration ID	RTN_FECFRAME size (bits)	Bit interleaver specification	
		N_{RTN}	M_{RTN}
RACH_ CR_{160} _ SF_{16} _ DB_{512}	1548	36	43
RACH_ CR_{160} _ SF_{4} _ DB_{2048}	6156	36	171
RACH_ CR_{160} _ SF_{16} _ DB_{288}	876	12	73
RACH_ CR_{160} _ SF_{4} _ DB_{976}	2940	28	105
Note: – N stands for the number of rows – M stands for the number of columns			

D018-COM-FUN-2090

The bit interleaver depth shall be 1 RTN_FECFRAME (1 Code Word).

D018-COM-FUN-2100

On the transmitter end, the encoded bits shall be serially written in column-wise and serially read out row-wise. MSB of RTN_FECFRAME is written and read out first.



D018-COM-FUN-2110

On the receiver end, the encoded bits shall be serially written in row-wise and serially read out column-wise. MSB of RTN_FECFRAME is written and read out first.

8.7.2.7 Auxiliary channel generation

D018-COM-FUN-2120

The Auxiliary Channel Generation module shall generate the Pilot sequence to support the channel estimation for coherent detection.

D018-COM-FUN-2130

The output stream of the Auxiliary Channel Generation shall be RTN_AUXFRAME. The RTN_AUXFRAME consists of a known pilot sequence of N_{RTN_PB} bits.

D018-COM-FUN-2140

The number of pilot bits (N_{RTN_PB}) of the Auxiliary channel shall depend on the RTN burst configuration and be as reported in the following table.

RACH Configuration ID	N_{RTN_PB} (bits)
$RACH\_CR_{160}\_SF_{16}\_DB_{512}$	1548 bits
$RACH\_CR_{160}\_SF_{4}\_DB_{2048}$	6156 bits
$RACH\_CR_{160}\_SF_{16}\_DB_{288}$	876 bits
$RACH\_CR_{160}\_SF_{4}\_DB_{976}$	2940 bits

D018-COM-FUN-2150

The contents of the Auxiliary channel (RTN_AUXFRAME) shall be generated using a binary m-sequence x.

The binary m-sequence x is generated by means of a generator polynomial of degree 15.

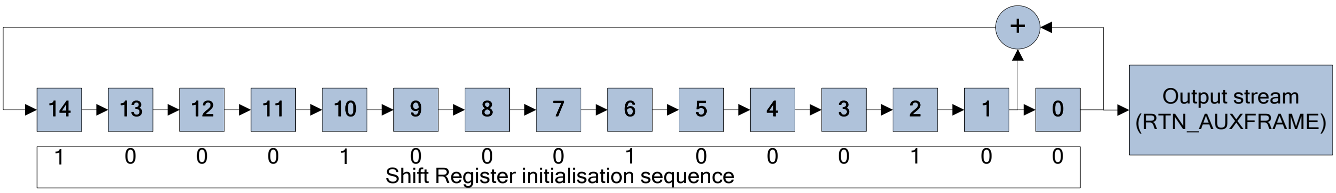
The x sequence is constructed using the primitive (over GF(2)) polynomial $x^{15} + x + 1$.

Let x(i) denote the i-th bit of the sequence.

The m-sequence x is constructed as:

- Initial conditions: x(14,...,0) = [1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0]
- Recursive definition of subsequent bits: x(i+15) = x(i+1) + x(i) modulo 2, i=0,...,N_{RTN_PB}-15.

The binary sequence generation is depicted in the following figure.

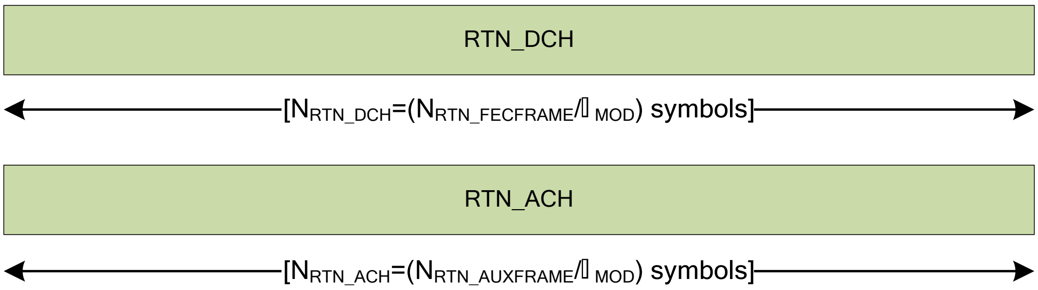


8.7.2.8 *Bit mapping into constellation*

The bit mapping module shall perform the mapping of the incoming bits into symbols according to the specified constellation order.

D018-COM-FUN-2170

The input stream of the Bit mapping module shall be a RTN_FECFRAME or a RTN_AUXFRAME and the output stream a RTN_DCH (Data Channel) or a RTN_ACH (Auxiliary Channel), respectively.



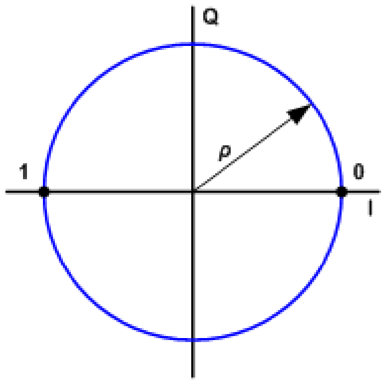
D018-COM-FUN-2180

The RTN_FECFRAME and the RTN_AUXFRAME shall be bit mapped independently into BPSK modulation (η_{MOD}=1) according to D018-COM-FUN-2190 below.

D018-COM-FUN-2190

The bit mapping into the BPSK constellation shall be compliant with the figure below.

One bit of the RTN_FECFRAME is mapped to a 1 BPSK symbol. One bit of the RTN_AUXFRAME is mapped to a 1 BPSK symbol.



8.7.2.9 Spreading

D018-COM-FUN-2200

The Spreading module shall perform

- the channelization of both channels (Data and Auxiliary channel)
- the complex scrambling

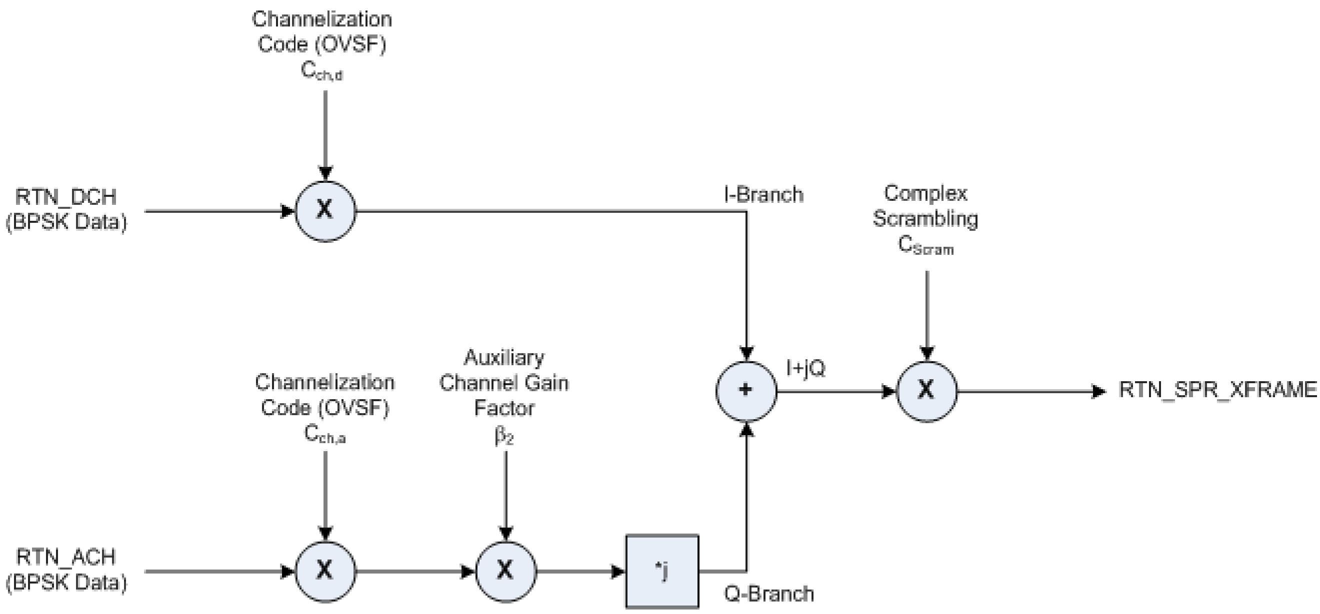
D018-COM-FUN-2210

The input stream of the Spreading module shall be a RTN_DCH (I component) and a RTN_ACH (Q component) and the output stream a RTN_SPR_XFRAME.

D018-COM-FUN-2220

The spreading of RTN_DCH and RTN_ACH shall be applied as illustrated in the following figure. It consists of two operations:

1. The channelization operation, which transforms every symbol into a number of chips, increasing the bandwidth of the signal ($C_{ch,d}$ and $C_{ch,a}$ channelization codes).
2. The scrambling operation, where a complex scrambling code is applied to the spread signal (C_{Scram} scrambling code).



D018-COM-FUN-2230

The RTN_DCH and RTN_ACH shall be spread to the chip rate by the channelization codes $C_{ch,d}$ and $C_{ch,a}$, respectively. The channelization codes used are OVSF codes following the specification described in section 8.7.2.9.1.

D018-COM-FUN-2240

After the channelization, the Auxiliary part (RTN_ACH) shall be weighted by the gain factor

$$\beta_2 = \sqrt{0.1}.$$

D018-COM-FUN-2250

The stream of the real-valued chips on the I-branch (RTN_DCH) and Q-branch (RTN_ACH) shall be treated as a complex-valued stream of chips, the RTN_DCH (I component) being the real part and the RTN_ACH (Q component) the imaginary part.

D018-COM-FUN-2260

The complex-valued stream (I+jQ) shall be scrambled by the complex valued scrambling code C_{scram}, (complex product). The scrambling code is specified in section 8.7.2.9.2.

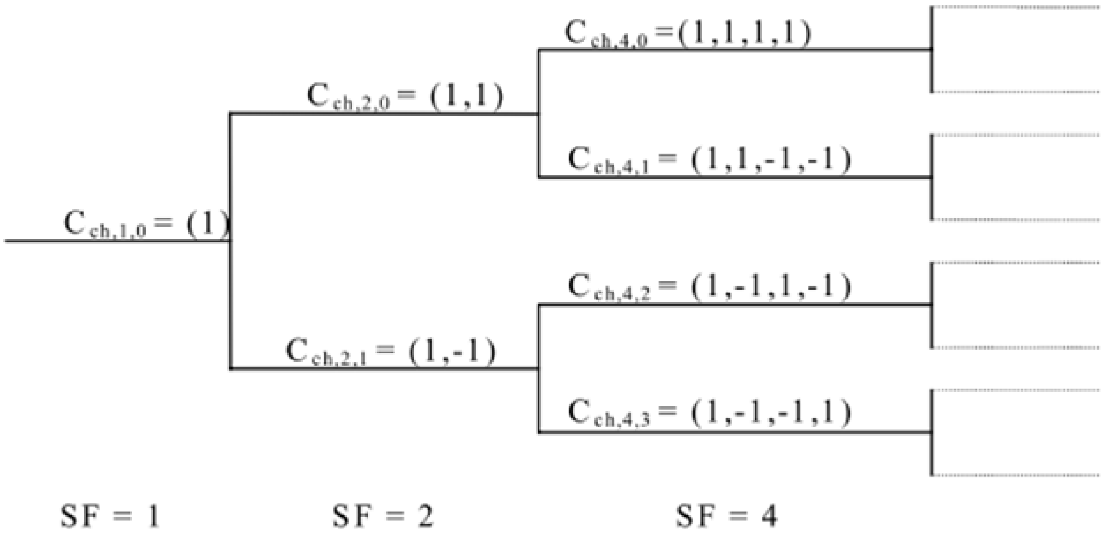
8.7.2.9.1 Channelization

D018-COM-FUN-2270

The channelization codes ($C_{ch,d}$, $C_{ch,a}$) shall be Orthogonal Variable Spreading Factor (OVSF) codes. The channelization codes are used to distinguish the RTN_DCH from the RTN_ACH.

D018-COM-FUN-2280

The OVSF codes shall be generated using the code tree of the following figure.



Channelization codes are uniquely described as $C_{ch,SF,k}$, where SF is the spreading factor of the code and k is the code number, 0<=k<=SF-1.The generation method for the channelization code is defined as:

$$C_{ch,1,0} = 1$$

$$\begin{bmatrix} C_{ch,2,0} \\ C_{ch,2,1} \end{bmatrix} = \begin{bmatrix} C_{ch,1,0} & C_{ch,1,0} \\ C_{ch,1,0} & -C_{ch,1,0} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$\begin{bmatrix} C_{ch,2^{(n+1)},0} \\ C_{ch,2^{(n+1)},1} \\ C_{ch,2^{(n+1)},2} \\ C_{ch,2^{(n+1)},3} \\ \vdots \\ C_{ch,2^{(n+1)},2^{(n+1)}-2} \\ C_{ch,2^{(n+1)},2^{(n+1)}-1} \end{bmatrix} = \begin{bmatrix} C_{ch,2^n,0} & C_{ch,2^n,0} \\ C_{ch,2^n,0} & -C_{ch,2^n,0} \\ C_{ch,2^n,1} & C_{ch,2^n,1} \\ C_{ch,2^n,1} & -C_{ch,2^n,1} \\ \vdots & \vdots \\ C_{ch,2^n,2^n-1} & C_{ch,2^n,2^n-1} \\ C_{ch,2^n,2^n-1} & -C_{ch,2^n,2^n-1} \end{bmatrix}$$

D018-COM-FUN-2290

The RTN_DCH shall be spread by the following channelization codes ($C_{ch,d}$), depending on the Burst Configuration ID, as reported in the following table. Channelization codes ($C_{ch,d}$) are uniquely identified by the code number k_d for each Burst Configuration.

Configuration ID	Channelization code length	Channelization code allocation ($C_{ch,d}$)
RACH_CR$_{160}$_SF$_{16}$_DB$_{512}$	16 chips	$C_{ch,16,kd}$, $0 \le kd \le 15$
RACH_CR$_{160}$_SF$_{4}$_DB$_{2048}$	4 chips	$C_{ch,4,kd}$, $0 \le kd \le 3$
RACH_CR$_{160}$_SF$_{16}$_DB$_{288}$	16 chips	$C_{ch,16,kd}$, $0 \le kd \le 15$
RACH_CR$_{160}$_SF$_{4}$_DB$_{976}$	4 chips	$C_{ch,4,kd}$, $0 \le kd \le 3$

D018-COM-FUN-2300

The RTN_ACH shall be spread by the following channelization codes ($C_{ch,a}$), depending on the Burst Configuration ID, as reported in the following table. Channelization codes ($C_{ch,a}$) are uniquely identified by the code number k_a for each Burst Configuration.

Configuration ID	Channelization code length	Channelization code allocation ($C_{ch,a}$)
RACH_CR$_{160}$_SF$_{16}$_DB$_{512}$	16 chips	$C_{ch,16,ka}$, $0 \le ka \le 15$
RACH_CR$_{160}$_SF$_{4}$_DB$_{2048}$	4 chips	$C_{ch,4,ka}$, $0 \le ka \le 3$
RACH_CR$_{160}$_SF$_{16}$_DB$_{288}$	16 chips	$C_{ch,16,ka}$, $0 \le ka \le 15$
RACH_CR$_{160}$_SF$_{4}$_DB$_{976}$	4 chips	$C_{ch,4,ka}$, $0 \le ka \le 3$

Note: the RTN_ACH uses the same SF as the RTN_DCH.

8.7.2.9.2 Complex Scrambling

D018-COM-FUN-2310

The scrambling code C_{scram} shall be a complex-valued sequence.

D018-COM-FUN-2320

The scrambling code C_{scram} shall be generated from two real-valued sequences $C_{1,n}$ and $C_{2,n}$., as defined in D018-COM-FUN-2330 below.

A scrambling code is uniquely identified by a 24-bit number n, which identifies the real-valued sequences $C_{1,n}$ and $C_{2,n}$.

D018-COM-FUN-2330

The real-valued sequences $C_{1,n}$ and $C_{2,n}$ are constructed from position-wise modulo-2 sum of segments of two binary m-sequences generated by means of a generator polynomial of degree 25. Let x and y be the two m-sequences respectively. The x sequence is constructed using the primitive (over GF(2)) polynomial $x^{25} + x^3 + 1$. The y sequence is constructed using the polynomial $x^{25} + x^3 + x^2 + x + 1$. The resulting sequences thus constitute segments of a set of Gold sequences.

The sequence $C_{2,n}$ is a 16 777 232 chip shifted version of the sequence $C_{1,n}$.

Let n_{23} ... n_0 be the 24 bit binary representation of the scrambling sequence number n with n_0 being the least significant bit. The x sequence depends on the chosen scrambling sequence number n and is denoted x_n in the sequel. Furthermore, let $x_n(i)$ and y(i) denote the i-th bit of the sequence x_n and y, respectively.

The m-sequences x_n and y are constructed as:

- Initial conditions:

  $x_n(0) = n_0$, $x_n(1)=n_1$, ..., $x_n(23)=n_{23}$, $x_n(24)=1$;

  $y(0) = y(1) = ... = y(23) = y(24) = 1$.

- Recursive definition of subsequent bits:

  $x_n(i+25) = x_n(i+3) + x_n(i)$ modulo 2, $i=0,...,2^{25}-27$;

  $y(i+25) = y(i+3) + y(i+2) + y(i+1) + y(i)$ modulo 2, $i=0,...,2^{25}-27$.

Define the binary Gold sequence z_n by:

$z_n(i) = x_n(i) + y(i)$ modulo 2, $i=0,1,2,...,2^{25}-2$

The real-valued Gold sequence Z_n is defined by:

$$Z_n(i) = \begin{cases} +1 \; if \; z_n(i) = 0 \\ -1 \; if \; z_n(i) = 1 \end{cases} \quad for \; i = 0,1,\dots,2^{25}-2$$

Now, the real-valued sequences $C_{1,n}$ and $C_{2,n}$ are defined as follows:

$$C_{1,n}(i) = Z_n(i), \;\; i = 0,1,2,\dots,2^{25}-2 \;\; and$$

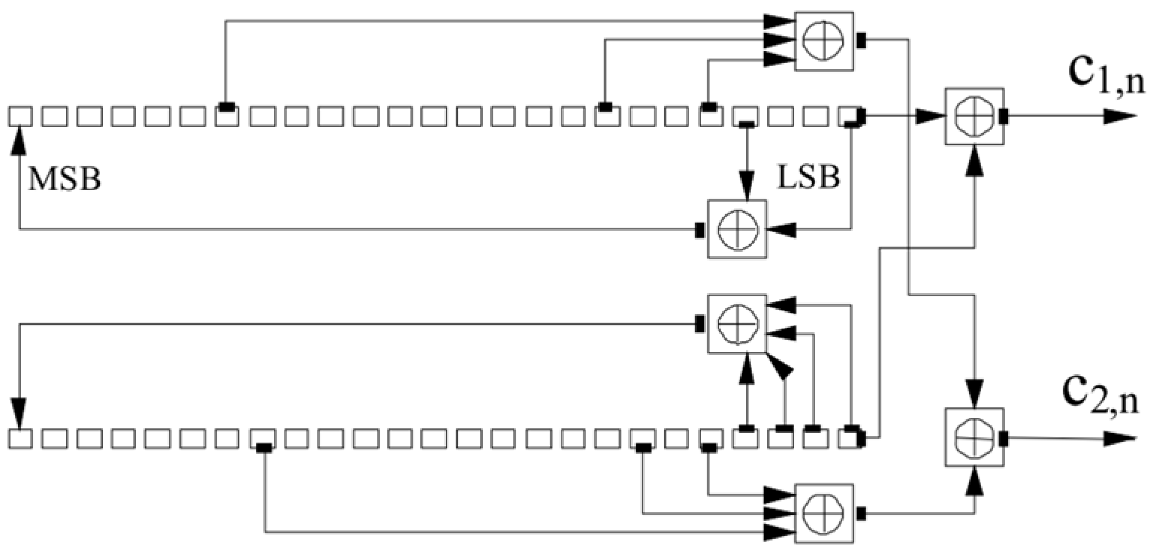
$$C_{2,n}(i) = Z_n\big((i + 16\,777\,232) \; modulo \; (2^{25}-1)\big), i = 0,1,2,\dots,2^{25}-2.$$

Code construction for $C_{1,n}$ and $C_{2,n}$ is shown in the figure below.

Finally, the complex-valued scrambling sequence C_{scram}, is defined as:

$$C_{scram}(i) = \frac{1}{\sqrt{2}} \cdot C_{1,n}(i) \cdot \left[1 + j \cdot (-1)^i \cdot C_{2,n}(2 \cdot \lfloor i/2 \rfloor)\right], \;\; i = 0,1,\dots,2^{25}-2$$

where $\lfloor . \rfloor$ denotes rounding to nearest lower integer.



D018-COM-FUN-2340

The scrambling sequence shall be truncated to the data channel size in chips. The size of the scrambling sequence is:

RACH Configuration ID	Scrambling code length (chips)
RACH_CR$_{160}$_SF$_{16}$_DB$_{512}$	24,768
RACH_CR$_{160}$_SF$_{4}$_DB$_{2048}$	24,624
RACH_CR$_{160}$_SF$_{16}$_DB$_{288}$	14,016
RACH_CR$_{160}$_SF$_{4}$_DB$_{976}$	11,760

8.7.2.10 Preamble generation

D018-COM-FUN-2350

The RTN_PREAMBLE shall consist of $N_{RTN\_PREAMBLE}$ known complex symbols. The preamble length is as reported in the following table.

RACH Configuration ID	$N_{RTN\_PREAMBLE}$ (symbols)
RACH_CR$_{160}$_SF$_{16}$_DB$_{512}$	128 symbols
RACH_CR$_{160}$_SF$_{4}$_DB$_{2048}$	128 symbols
RACH_CR$_{160}$_SF$_{16}$_DB$_{288}$	128 symbols
RACH_CR$_{160}$_SF$_{4}$_DB$_{976}$	128 symbols

D018-COM-FUN-2360

The complex-valued preamble sequence RTN_PREAMBLE shall be generated from a real-valued Gold sequence Z_n, the sequence RTN_PREAMBLE being uniquely identified by the 9-bit index n (n=0, 1, ..., 2^9-2), which identifies the Gold sequence.

The complex-valued preamble sequence is constructed from position-wise modulo-2 sum of $N_{RTN\_PREAMBLE}$ chip segments of the Gold code Z_n.

Two binary m-sequences generated by means of a generator polynomial of degree 9 are used to construct a binary Gold code z. Let x and y be the two m-sequences respectively. The x sequence is constructed using the primitive (over GF(2)) polynomial $x^9 + x^4 + 1$. The y sequence is constructed using the polynomial $x^9 + x^4 + x^3 + x + 1$.

The Gold sequence z is actually a function of the chosen sequence index n and is thus denoted z_n in the sequel. Furthermore, let x(i), y(i) and z_n(i) denote the i-th symbol of the sequence x, y and z_n respectively.

The m-sequences x and y are constructed as:

- Initial conditions:

  x(0) = 1, x(1) = x(2) =, ...,= x(7) = x(8) = 0;

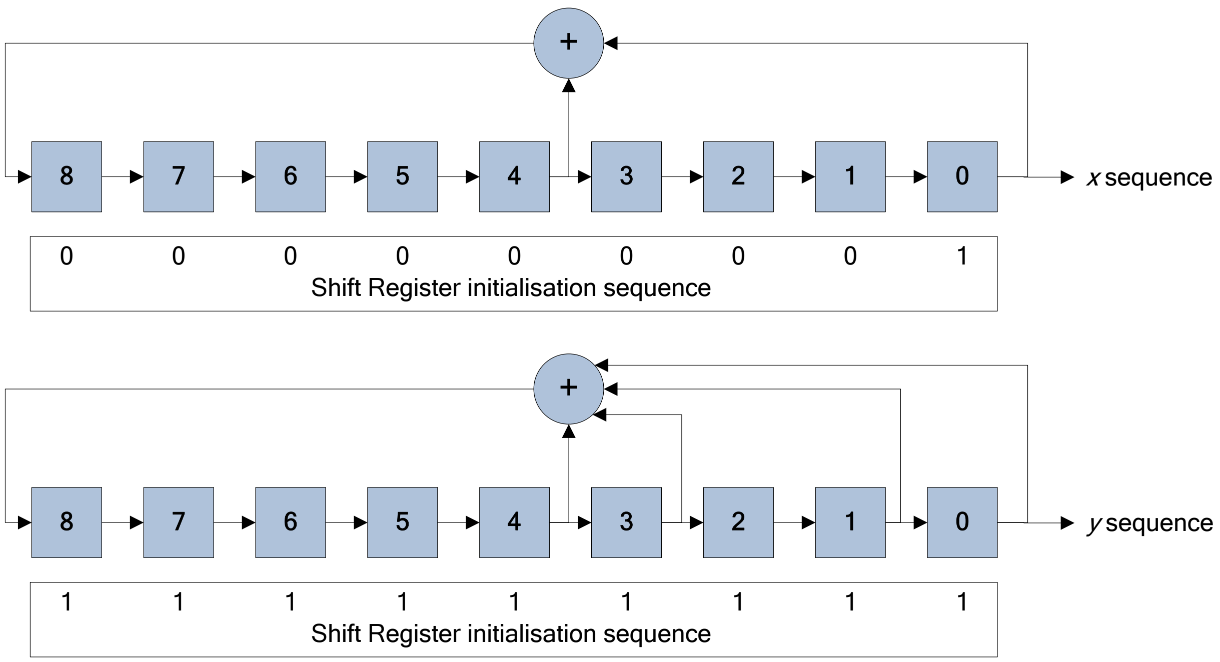
  y(0) = y(1) = ... = y(7) = y(8) = 1.

- Recursive definition of subsequent bits:

  x(i+9) = x(i+4) + x(i) modulo 2, i=0,...,2^9-11;

  y(i+9) = y(i+4) + y(i+3) + y(i+1) + y(i) modulo 2, i=0,...,2^9-11.

x and y sequences construction is depicted in the following figure.



Define the binary Gold sequence z_n by:

$z_n(i) = x((i+n) \text{ modulo } 2^9-1) + y(i) \text{ modulo } 2$, $i=0,1,2,...,2^9-2$

The real-valued Gold sequence Z_n is defined by:

$$Z_n(i) = \begin{cases} +1 \text{ } if \text{ } z_n(i) = 0 \\ -1 \text{ } if \text{ } z_n(i) = 1 \end{cases} \quad for \text{ } i = 0,1,\dots,2^9-2$$

Finally, the complex-valued preamble sequence RTN_PREAMBLE is defined as:

$$RTN\_PREAMBLE(i) = \frac{1}{\sqrt{2}} \cdot [Z_n(i) + j \cdot Z_n(i+256)], \quad i = 0,1,\dots,N_{RTN\_PREAMBLE} - 1$$

8.7.2.11 *Preamble spreading*

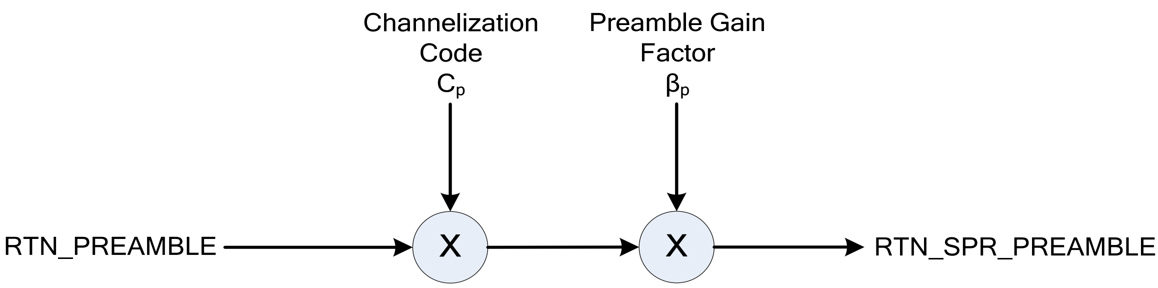
D018-COM-FUN-2370

The input stream of the Preamble Spreading module shall be a RTN_PREAMBLE and the output stream a RTN_SPR_PREAMBLE.

D018-COM-FUN-2380

The spreading of RTN_PREAMBLE shall be applied as illustrated in the following figure. It consists of one operation:

1. The channelization operation, which transforms every preamble complex symbol into a number of complex chips, increasing the bandwidth of the signal (C_p channelization code).



D018-COM-FUN-2390

The RTN_PREAMBLE shall be spread to the chip rate by the channelization code C_p. The channelization codes used are:

RACH Configuration ID	Channelization code length	Preamble channelization code allocation (C_p)
RACH_CR$_{160}$_SF$_{16}$_DB$_{512}$	16	[1, j, j, 1, j, -1, -1, j, j, 1, -1, -j, 1, -j, j, -1]·exp(jπ/4)

$RACH\_CR_{160}\_SF_4\_DB_{2048}$	4	[1, -j, j, -1]·exp(jπ/4)
$RACH\_CR_{160}\_SF_{16}\_DB_{288}$	16	[1, j, j, 1, j, -1, -1, j, j, 1, -1, -j, 1, -j, j, -1]·exp(jπ/4)
$RACH\_CR_{160}\_SF_4\_DB_{976}$	4	[1, -j, j, -1]·exp(jπ/4)

Note: the RTN_PREAMBLE uses the same SF as the RTN_DCH and RTN_ACH.

D018-COM-FUN-2400

After preamble channelization, the preamble shall be weighted by the gain factor

$$\beta_p = \rho\sqrt{1+\beta_2^2}$$

so that the average power of RTN_SPR_PREAMBLE is the same as the RTN_SPR_XFRAME, i.e., $\rho^2(1+\beta_2^2)$.

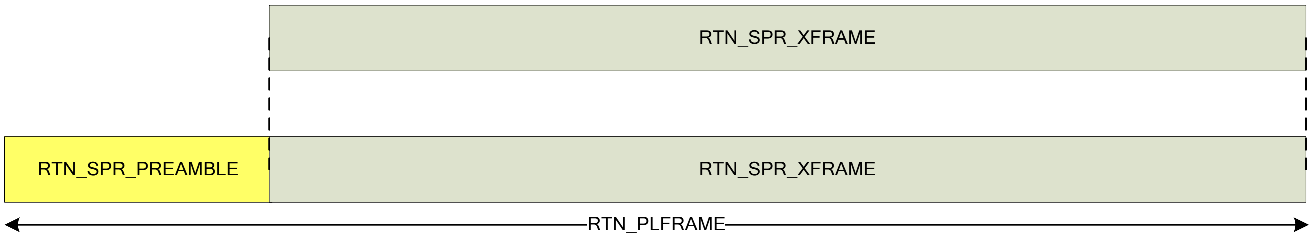
8.7.2.12 Physical Layer Framing

D018-COM-FUN-2410

The Physical Layer Framing module shall perform the RTN_SPR_PREAMBLE insertion.

D018-COM-FUN-2420

The input stream of the Physical Layer Framing module shall be a RTN_SPR_XFRAME and a RTN_SPR_PREAMBLE and the output a RTN_PLFRAME, as detailed in the following figure.



8.7.2.13 Base-band pulse shaping and quadrature modulation

D018-COM-FUN-2430

I and Q signals shall be shaped using a Square Root Raised Cosine with a roll-off factor α=0.2.

D018-COM-FUN-2440

The base-band SRRC filter shall have a theoretical function defined by the following expression.

$$H(f) = 1 \qquad \text{for } |f| < f_N(1-\alpha)$$

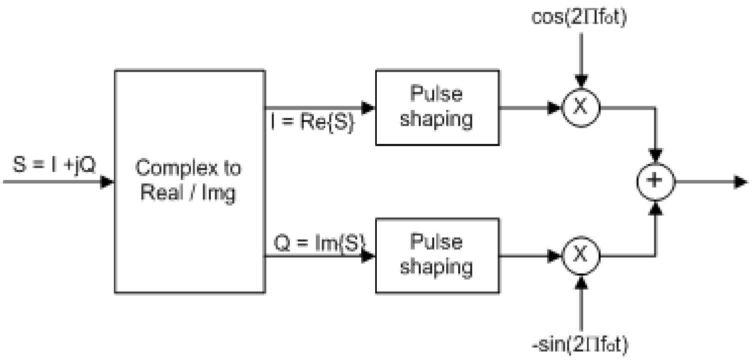
$$H(f) = \left\{ \frac{1}{2} + \frac{1}{2}\sin\frac{\pi}{2f_N}\left[\frac{f_N - |f|}{\alpha}\right]\right\}^{\frac{1}{2}} \qquad \text{for } f_N(1-\alpha)$$

$$H(f) = 0 \ \text{ for } |f| > f_N(1+\alpha),$$

where $f_N = R_s/2$ is the Nyquist frequency and α is the roll-off factor.

D018-COM-FUN-2450

The quadrature modulation shall be performed by multiplying the In-phase component (I) by $\cos(2\pi f_0 t)$ and the quadrature component (Q) by $-\sin(2\pi f_0 t)$. The resulting I and Q components are added to conform the modulator output signal.



8.7.3 Splitting traffic profiles

D018-COM-FUN-2470

The messages shall be mapped to the RACH burst Configurations based on message service requirements as detailed in the following table.

RACH Configuration ID	Service	Splitting policy
RACH_CR$_{160}$_SF$_4$_DB$_{976}$	Voice	Voice
RACH_CR$_{160}$_SF$_4$_DB$_{2048}$	Data/Voice	Message size ≥ 500 bytes Voice
RACH_CR$_{160}$_SF$_{16}$_DB$_{288}$	Data/Signalling	Messages or signalling that can be transmitted without fragmentation.
RACH_CR$_{160}$_SF$_{16}$_DB$_{512}$	Data/Signalling	Remaining messages or signalling

9. CONTROL PLANE SPECIFICATION

9.1 Control plane description

The Control plane deals with real-time control functionality that is required in order for the communication system to function, including:

- terminal registration (logon / logoff)
- handover
- synchronisation
- ACM
- radio resource management
- support to redundancy and failure detection
- security control

Refer to section 5.2 for details on the control plane.

9.2 Control procedures

Signalling messages mentioned in the following sections are specified in detail in section 11.

9.2.1 General

D018-COM-FUN-2480

The maximum number of retransmissions for all signalling messages except the LOGON REQUEST message shall be as indicated by the *generic_retransmission_number* configuration parameter.

Note: This parameter is provided in the LOGON INITIAL ACCEPT and HO COMMAND messages.

D018-COM-FUN-2481

When not specified otherwise by the signalling protocol, the retransmission time-out for the RTN link signalling messages shall be the '*retransmission_timeout'* value indicated in D018-COM-FUN-3410 associated with the '*CC_category_id*' with value '*Signalling*'.

D018-COM-FUN-2482

The maximum number of retransmissions for the LOGON REQUEST message shall be as indicated by the *Logon_request_retransmission_number* configuration parameter.

Note: This parameter is provided in the LOGON TABLE signalling message.

D018-COM-FUN-2490

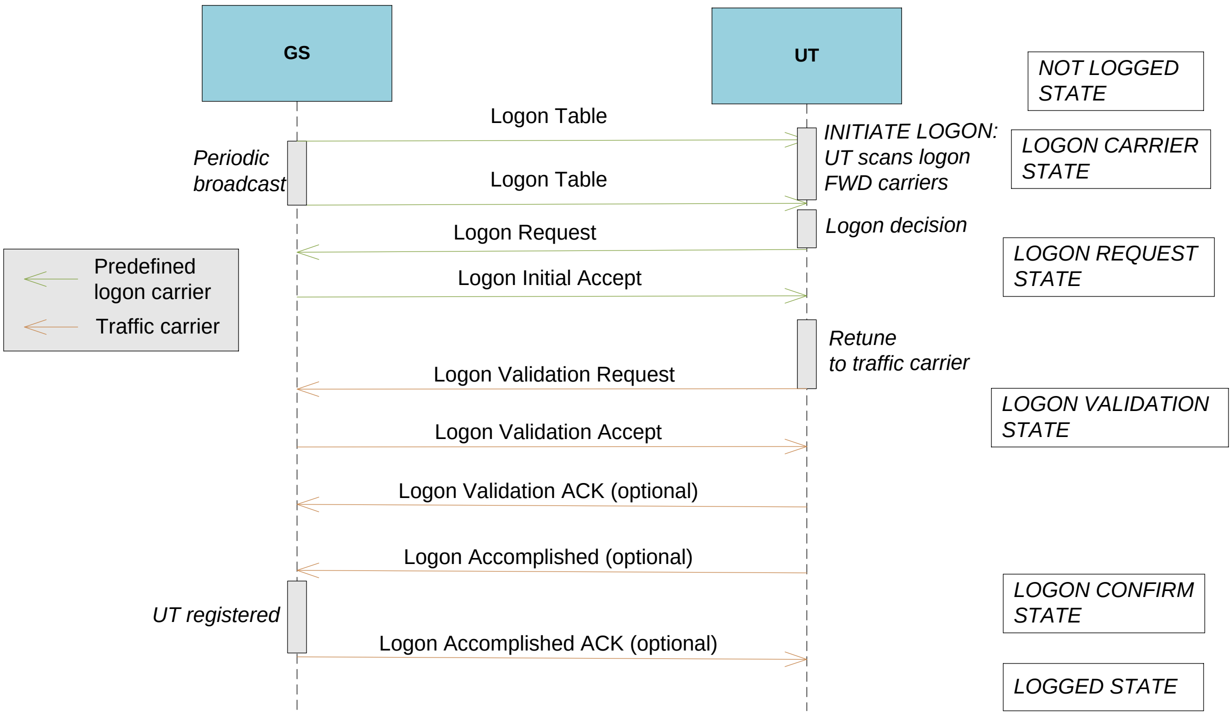
With respect to the protocols defined for all control procedures, use of link layer ARQ shall be disabled for the transmitted signalling messages.

9.2.2 Terminal registration procedure (logon/logoff)

9.2.2.1 Terminal logon procedure

D018-COM-FUN-2550

The following procedure shall be followed by the UT and GSE for the logon process:



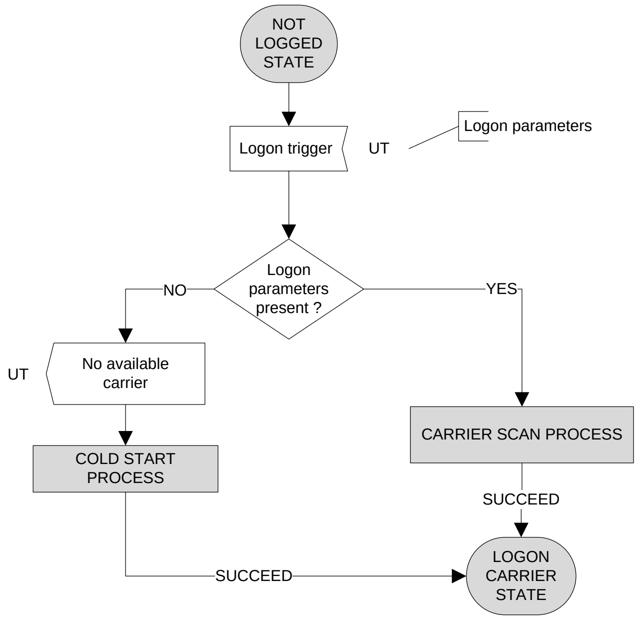
The procedure steps are detailed here:

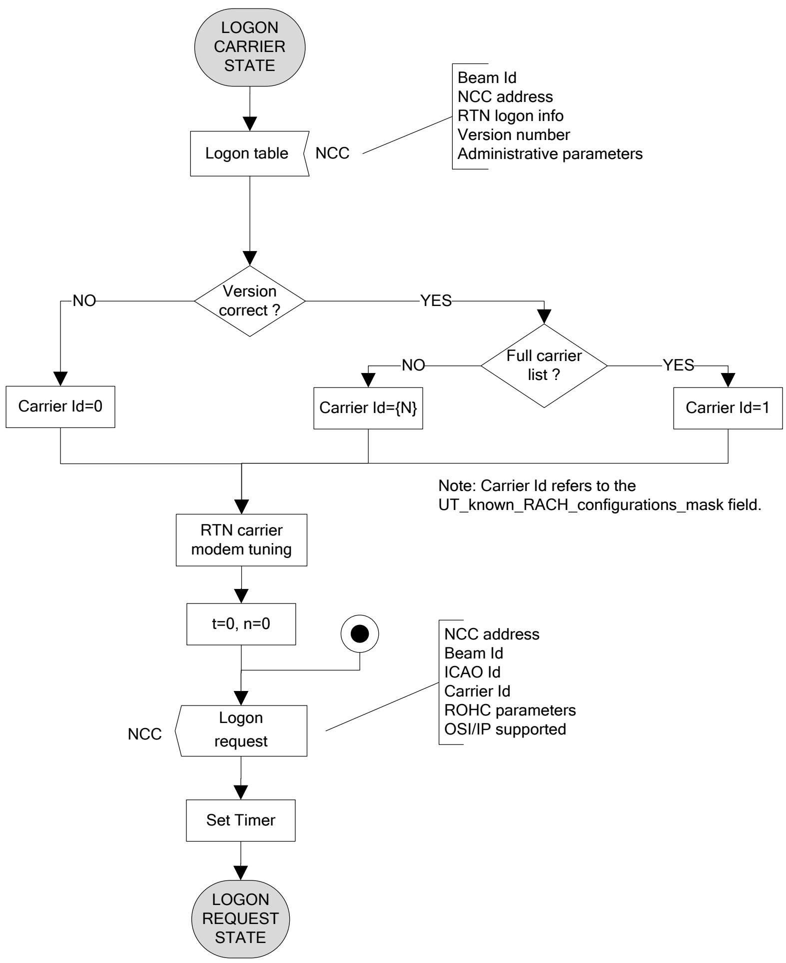
1. The GS transmits a broadcast carrier that contains the initial information (LOGON TABLE) in order for the UT to initiate its logon phase.
2. The UT receives this information, adjusts its modem parameters and transmits to the GS the LOGON REQUEST, which contains the UT preferences.
3. The GS answers to the UT by providing it the description of the RTN, FWD carrier and ground station to be used in the LOGON INITIAL ACCEPT.
4. Using the FWD/RTN link parameters indicated in the LOGON INITIAL ACCEPT message, the UT re-tunes its modem and sends a LOGON VALIDATION REQUEST message to the GS.

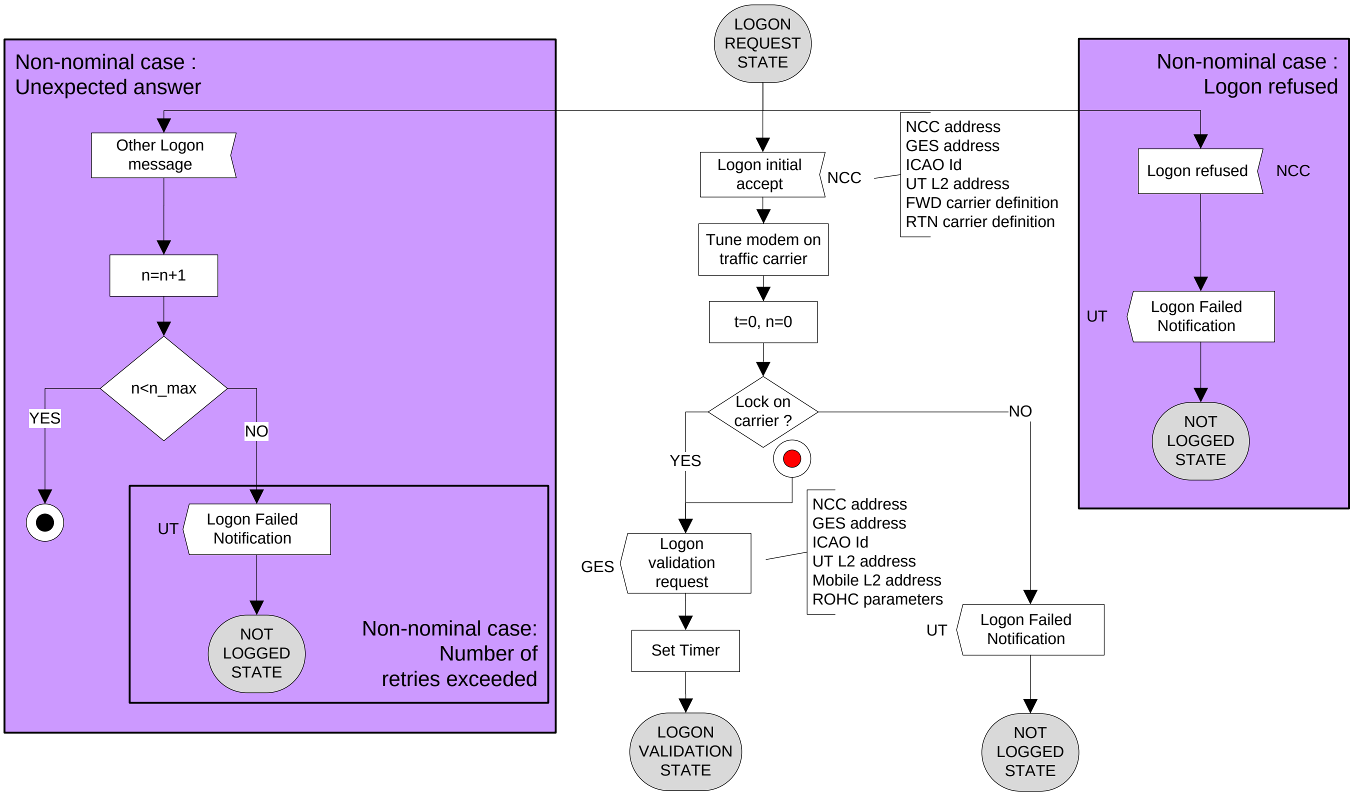
5. The GS validates the request and confirms with a LOGON VALIDATION ACCEPT message.
6. Optionally, if it is requested by GS, the UT confirms the reception of this message by sending a LOGON VALIDATION ACK message.
7. Optionally, if it is requested by GS, the UT confirms the success of the logon process by sending a LOGON ACCOMPLISHED message.
8. The GS acknowledges the reception by sending a LOGON ACCOMPLISHED ACK message

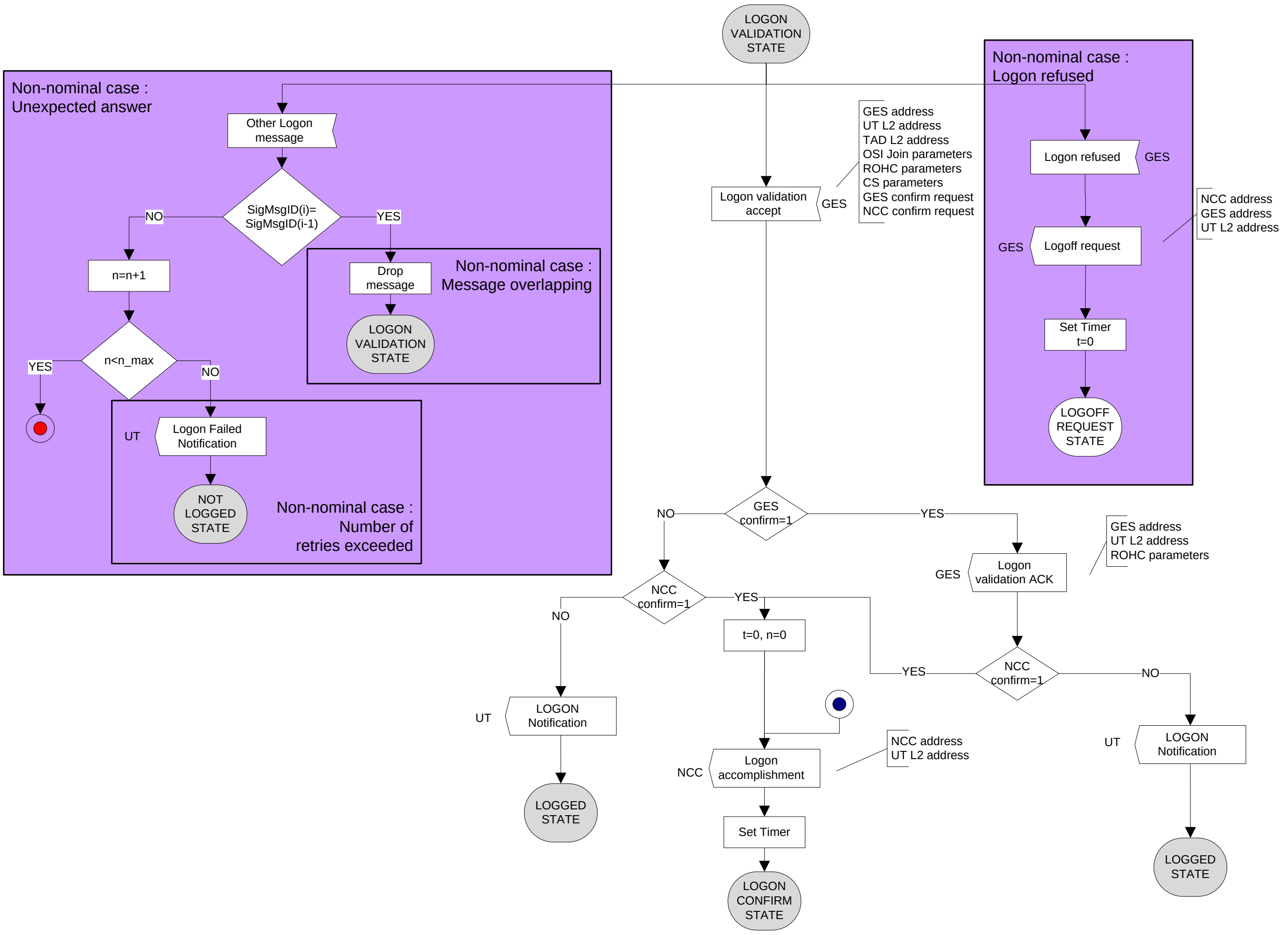
D018-COM-FUN-2551

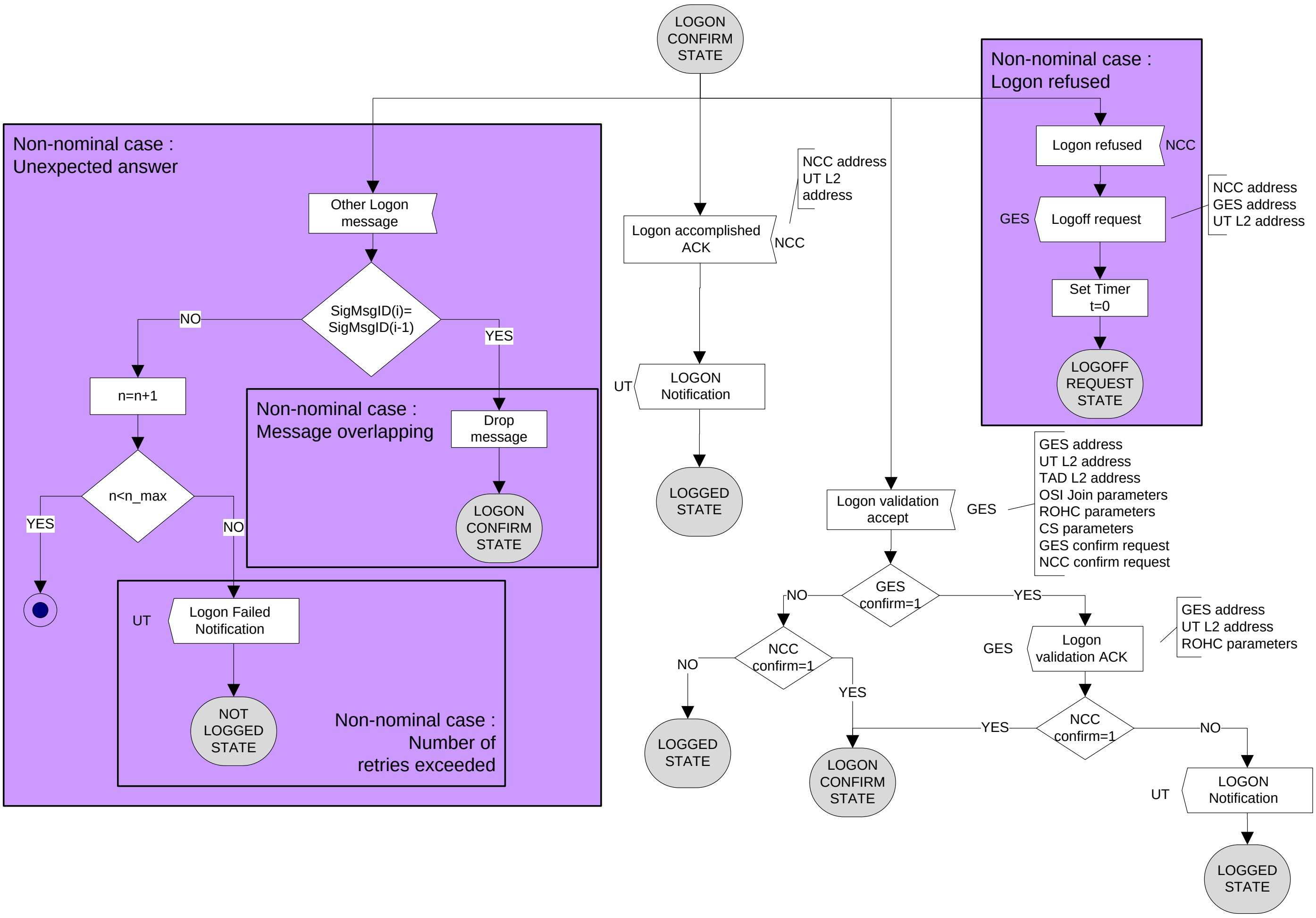
The following SDLs shall be followed by the UT for the logon process:




LOGON CARRIER STATE
Logon table
NCC
Beam Id
NCC address
RTN logon info
Version number
Administrative parameters
Version correct ?
NO
YES
Full carrier list ?
NO
YES
Carrier Id=0
Carrier Id={N}
Carrier Id=1
Note: Carrier Id refers to the UT_known_RACH_configurations_mask field.
RTN carrier modem tuning
t=0, n=0
NCC address
Beam Id
ICAO Id
Carrier Id
ROHC parameters
OSI/IP supported
NCC
Logon request
Set Timer
LOGON REQUEST STATE


LOGON REQUEST STATE
Non-nominal case : Unexpected answer
Other Logon message
n=n+1
n<n_max
YES
NO
UT
Logon Failed Notification
NOT LOGGED STATE
Non-nominal case: Number of retries exceeded
Logon initial accept
NCC
NCC address
GES address
ICAO Id
UT L2 address
FWD carrier definition
RTN carrier definition
Tune modem on traffic carrier
t=0, n=0
Lock on carrier ?
YES
NO
GES
Logon validation request
NCC address
GES address
ICAO Id
UT L2 address
Mobile L2 address
ROHC parameters
Set Timer
LOGON VALIDATION STATE
UT
Logon Failed Notification
NOT LOGGED STATE
Non-nominal case : Logon refused
Logon refused
NCC
UT
Logon Failed Notification
NOT LOGGED STATE


LOGON VALIDATION STATE
Non-nominal case : Unexpected answer
Other Logon message
SigMsgID(i)= SigMsgID(i-1)
NO
YES
n=n+1
Drop message
Non-nominal case : Message overlapping
LOGON VALIDATION STATE
n<n_max
YES
NO
UT
Logon Failed Notification
NOT LOGGED STATE
Non-nominal case : Number of retries exceeded
Logon validation accept
GES
GES address
UT L2 address
TAD L2 address
OSI Join parameters
ROHC parameters
CS parameters
GES confirm request
NCC confirm request
Non-nominal case : Logon refused
Logon refused
GES
Logoff request
NCC address
GES address
UT L2 address
Set Timer t=0
LOGOFF REQUEST STATE
GES confirm=1
Logon validation ACK
GES address
UT L2 address
ROHC parameters
NCC confirm=1
t=0, n=0
LOGON Notification
LOGGED STATE
Logon accomplishment
NCC
NCC address
UT L2 address
Set Timer
LOGON CONFIRM STATE



Notes:

- *NOT LOGGED STATE*: this represents the initial state of the UT. The UT is not logged to any system and could not communicate with the ground.
- *LOGON CARRIER STATE*: this represents the state where the UT is locked on the logon carrier and can start to retrieve the ground station information.
- *LOGON REQUEST STATE*: this represents the state where the UT has received the system information, has sent its logon request to the ground station and is waiting for ground station answer.
- *LOGON VALIDATION STATE*: this represents the state where the UT has received the information about the traffic carrier, has initiated the logon request on the GES using the RTN traffic carrier and is waiting for GES validation.
- *LOGON CONFIRM STATE*: in the event that the ground segment requests a validation by the UT to the NCC, the UT has sent its notification to the NCC and is waiting for its answer.
- *LOGGED STATE*: the UT is logged on the system and can start to exchange data with the ground segment.
- *LOGOFF REQUEST STATE*: in this state the UT has requested a logoff and is waiting for ground segment acknowledgement (see D018-COM-FUN-2611).

- *COLD START PROCESS*: this represents the process to be followed when the UT does not get information to initiate its logon phase (no carrier information) in order to retrieve this information by trying to detect if any carrier is available.
- *CARRIER SCAN PROCESS:* this represents the process to be followed to acquire one of the logon carriers known by the UT.
- *SigMsgID*: In order to identify whether the received message is similar to the previous one. If the SigMsgIDs are similar this means that the two consecutive messages are part of the same step in the process. In that case the last message is dropped and the next message is expected.
- The value *n_max* is the one indicated by the *generic_retransmission_number* provided in the LOGON INITIAL ACCEPT and HO COMMAND messages.

D018-COM-FUN-2552

The UT shall send a LOGON ACK message to the GS when a LOGON VALIDATION message containing a GES confirm request flag is received in the LOGGED STATE.

D018-COM-FUN-2553

The UT shall return to the NOT LOGGED STATE and notify the corresponding process in the UT about the failure in the logon process when the maximum number of retrials in any logon protocol message is exceeded.

D018-COM-FUN-2565

The message LOGON TABLE shall be sent with a minimum periodicity of once every 30 seconds over a logon carrier.

9.2.2.2 Terminal logoff procedure

D018-COM-FUN-2580

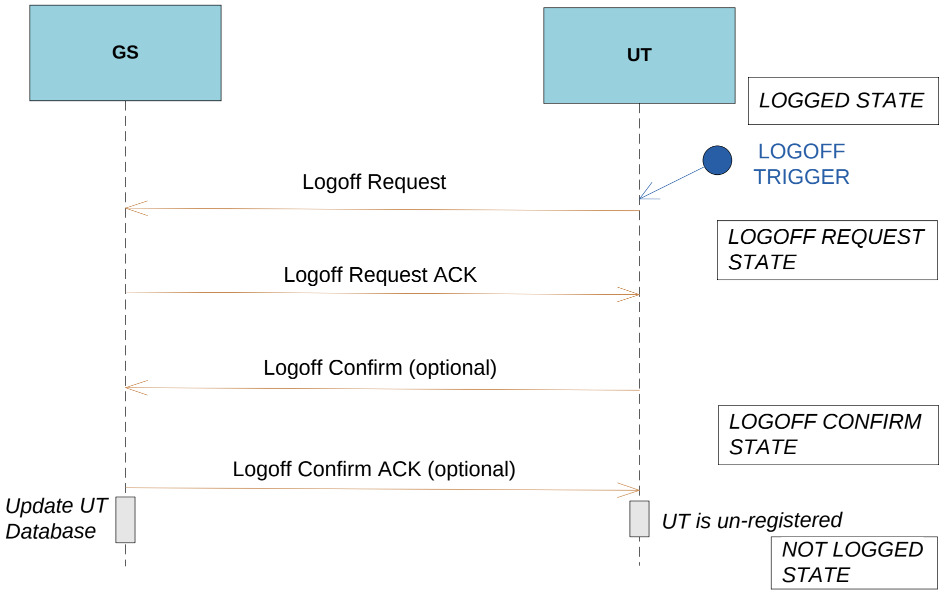
The UT shall be able to initiate a logoff procedure.

D018-COM-FUN-2590

The GS shall be able to trigger the logoff of a UT. The UT shall logoff at the request of the GS.

D018-COM-FUN-2600

The following procedure shall be followed by the UT and GSE for the air-initiated logoff process.

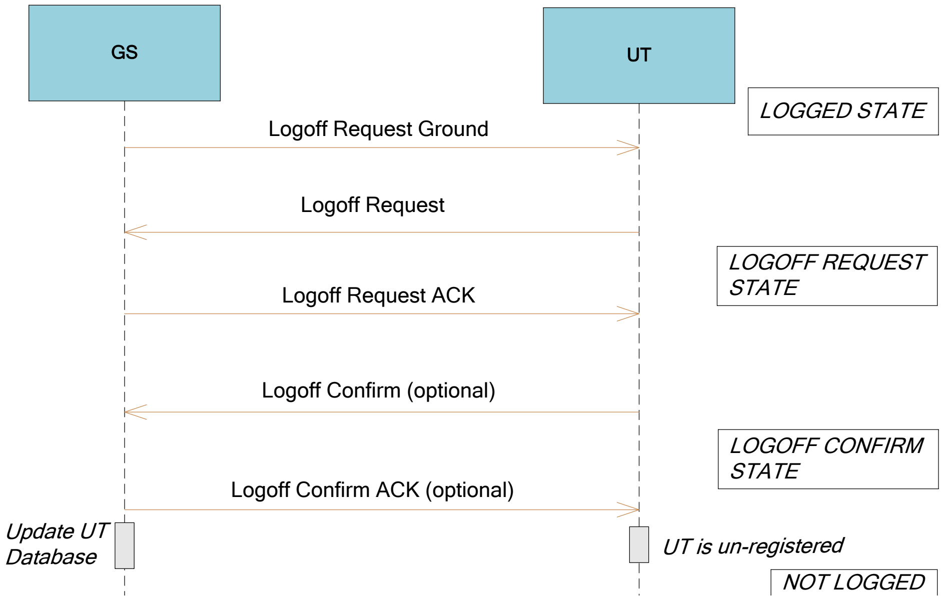


The procedure steps are detailed here:

1. The UT sends a LOGOFF REQUEST to the GS.
2. The GS acknowledges the logoff with a LOGOFF REQUEST ACK message.
3. Optionally, if it is requested by the GS, the UT confirms the logoff by sending a LOGOFF CONFIRM message.
4. The GS acknowledges the reception with the LOGOFF CONFIRM ACK message.

D018-COM-FUN-2610

The following procedure shall be followed by the UT and GSE for the ground-initiated logoff process:

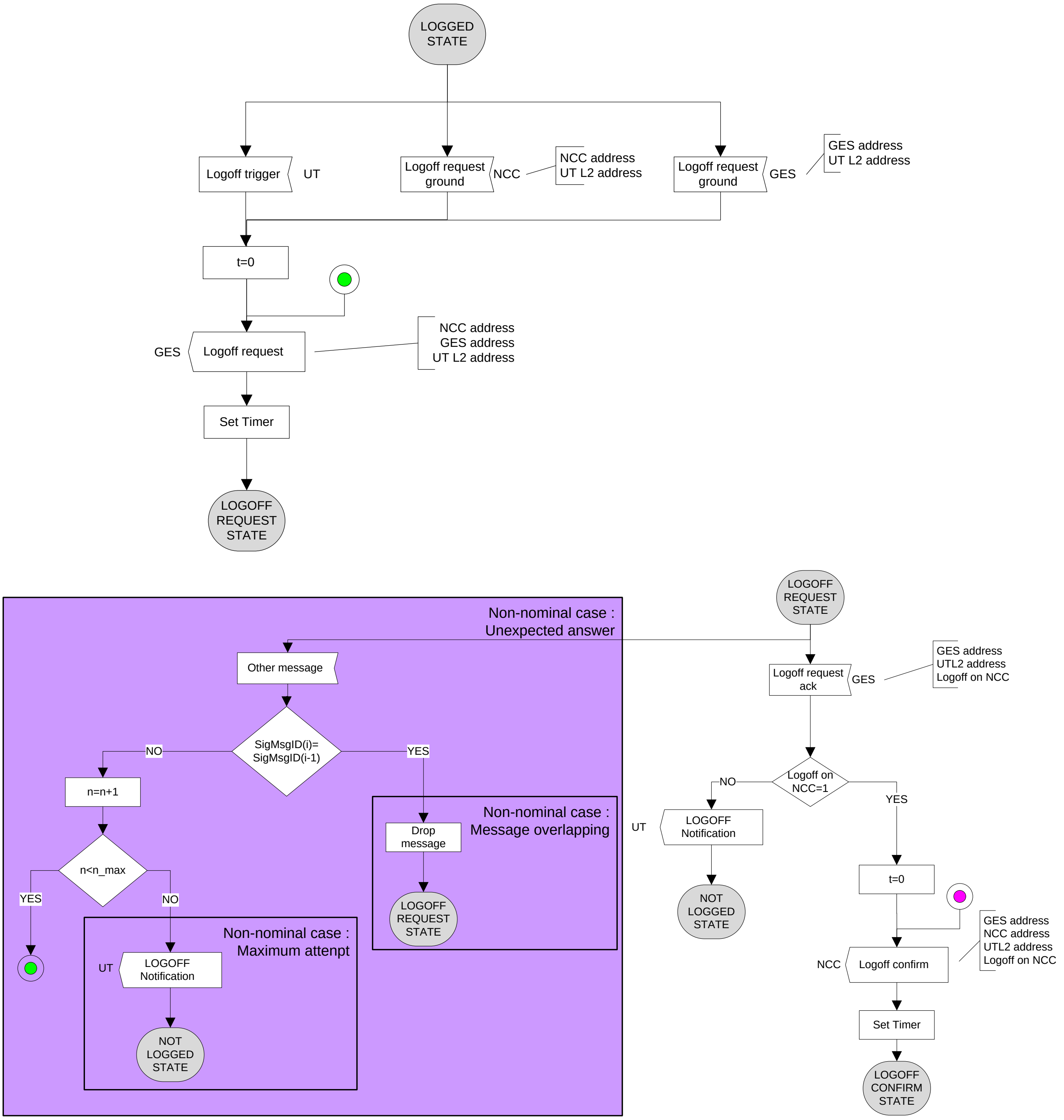


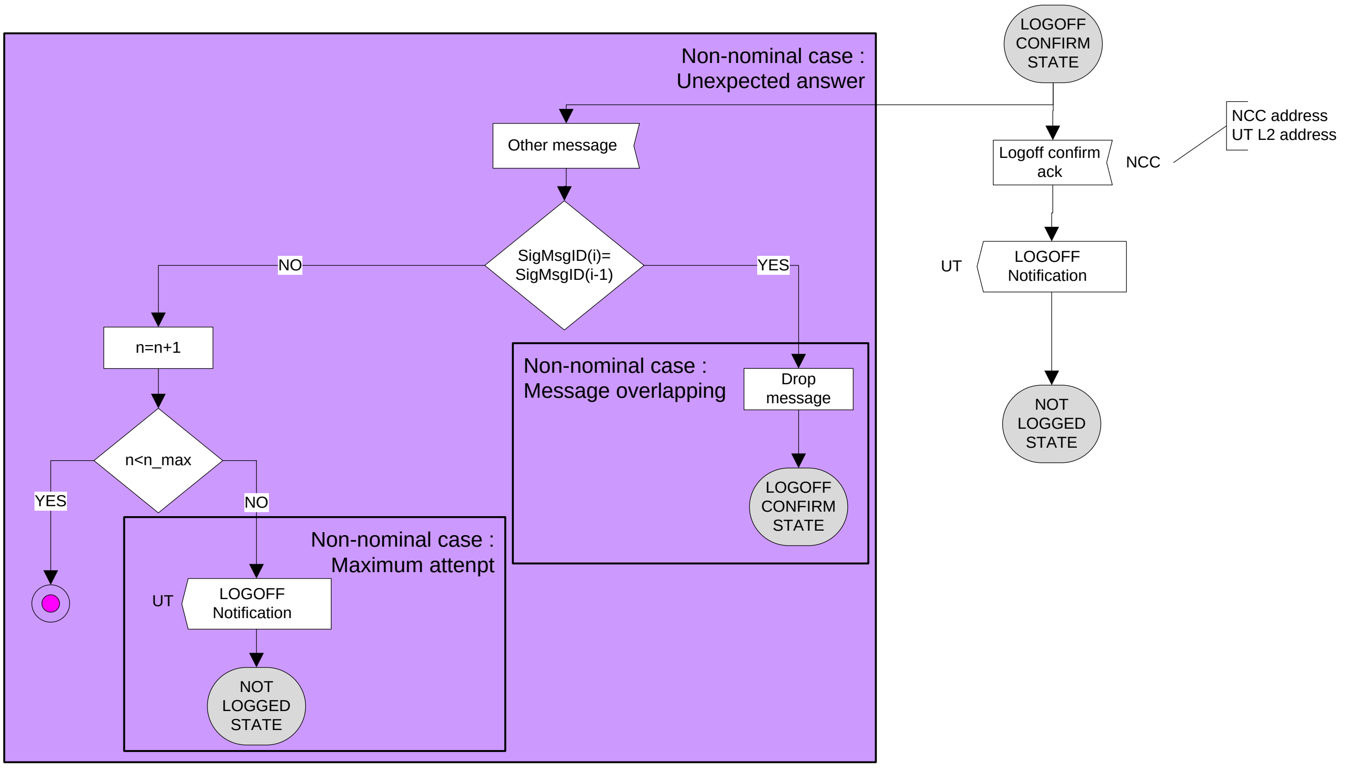
The procedure steps are detailed hereafter:

1. The GS sends a LOGOFF REQUEST - GROUND message to the UT.
2. The UT answers with a LOGOFF REQUEST message to the GS
3. The GS acknowledges the logoff with a LOGOFF REQUEST ACK message.
4. Optionally, if it is requested by GS, the UT confirms the logoff by sending a LOGOFF CONFIRM message.
5. The GS acknowledges the reception with the LOGOFF CONFIRM ACK message.

D018-COM-FUN-2611

The following SDLs shall be followed by the UT for the air-initiated and ground-initiated logoff processes.


LOGGED STATE
Logoff trigger
UT
Logoff request ground
NCC
NCC address
UT L2 address
Logoff request ground
GES
GES address
UT L2 address
t=0
GES
Logoff request
NCC address
GES address
UT L2 address
Set Timer
LOGOFF REQUEST STATE
Non-nominal case :
Unexpected answer
Other message
SigMsgID(i)=
SigMsgID(i-1)
NO
YES
n=n+1
n<n_max
YES
NO
Non-nominal case :
Maximum attempt
UT
LOGOFF Notification
NOT LOGGED STATE
Non-nominal case :
Message overlapping
Drop message
LOGOFF REQUEST STATE
LOGOFF REQUEST STATE
Logoff request ack
GES
GES address
UTL2 address
Logoff on NCC
Logoff on NCC=1
NO
YES
UT
LOGOFF Notification
NOT LOGGED STATE
t=0
NCC
Logoff confirm
GES address
NCC address
UTL2 address
Logoff on NCC
Set Timer
LOGOFF CONFIRM STATE



Notes:

- *LOGGED STATE*: the UT is logged onto the system and is able to communicate with the ground segment.
- *LOGOFF REQUEST STATE*: in this state the UT has requested a logoff and is waiting for ground segment acknowledgement.
- *NOT LOGGED STATE*: this represents the initial state of the UT. The UT is not logged onto any system and can not communicate with the ground.
- *LOGOFF CONFIRM STATE*: in the event that the ground segment has requested from the UT an acknowledgement on the NCC, the UT has notified the NCC about the logoff and is waiting for the acknowledgement.

D018-COM-FUN-2612

The UT shall return to the NOT LOGGED STATE when the maximum number of retrials in any logoff protocol message is exceeded.

D018-COM-FUN-2613

If a logged-on UT does not received any unicast data from its associated GS within a time frame of *No received data timeout* seconds, the UT shall (silently) go into logged-off state.

Note: This requirement allows recovery from a non-nominal situation where a UT considers itself logged-on while the GS considers it logged-off. In nominal conditions and if the UT generates no traffic, the GS may therefore have to transmit data to the UT to avoid unwanted expiration of the timer indicated above.

9.2.3 Handover procedures

9.2.3.1 General aspects

[REQUIREMENT DELETED]

9.2.3.2 Handover sequence

9.2.3.2.1 Handover detection

D018-COM-FUN-2630

The handover detection process in the UT shall be able to rely only on signal measurements.

D018-COM-FUN-2640

The UT shall constantly monitor the link quality of at least 3 neighbour/alternative channels (from same or different beams, satellites or SSPs) and compare the measurements with its current link quality in order to detect potential HO candidates.

D018-COM-FUN-2650

The UT shall have the capability to receive a second FWD carrier specifically devoted to handover purposes (handover detection and handover execution).

D018-COM-FUN-2680

The UT shall send a HO_RECOMMENDATION message to the GS when there is a neighbour channel with better quality than the current one, or when it is triggered by administrative, political or business related reasons, including manual HO triggered by the flight crew.

D018-COM-FUN-2690

The GS shall be able to receive and process HO_RECOMMENDATION messages from UTs or HO triggers generated by other GS subsystems (e.g., management module, redundancy module).

D018-COM-FUN-2691

The GS shall be able to send a HO_INFO_REQ message in any instant to request information of the link quality of current and neighbour/adjacent channels to the UT in order to use this information for the handover decision process (for GS-initiated handovers).

D018-COM-FUN-2692

The UT shall send a HO_RECOMMENDATION message to the GS after receiving a HO_INFO_REQ message.

9.2.3.2.2 Handover decision

D018-COM-FUN-2700

After receiving a HO_RECOMMENDATION message from the UT or any HO trigger from the GS and deciding the type and target channel and GS element, the GS shall send a HO_COMMAND message to the UT. At least the following HO types shall be identified:

HO Type	Description
1	Change of satellite service providers (SSP)
2	Beam/channel/satellite change within same GES and GES HO
3	Fast HO within same GES

D018-COM-FUN-2701

In addition to the HO types defined in D018-COM-FUN-2700, the UT shall be able to proceed to direct LOGON in the new SSP to perform a HO to a preferred SSP.

D018-COM-FUN-2702

The GS shall include in the HO_COMMAND message to the UT one flag to indicate if the HO procedure can be shortened by not sending the last messages corresponding to the HO_FINISHED and ACK_HO_FINISHED messages.

9.2.3.2.3 Handover execution

D018-COM-FUN-2710

A UT shall be capable of receiving simultaneously two FLCs (associated with same or different beams or different satellites).

D018-COM-FUN-2720

A UT shall be capable of transmitting not simultaneously RACH bursts on two different RLCs, applying the corresponding Doppler pre-compensation on a burst-by-burst basis for each RLC if different satellites are involved in the handover.

D018-COM-FUN-2740

A UT shall be able to maintain two sessions in parallel during HO execution, using different or the same satellite channels for Rx and Tx, and with different or the same GSE.

D018-COM-FUN-2750

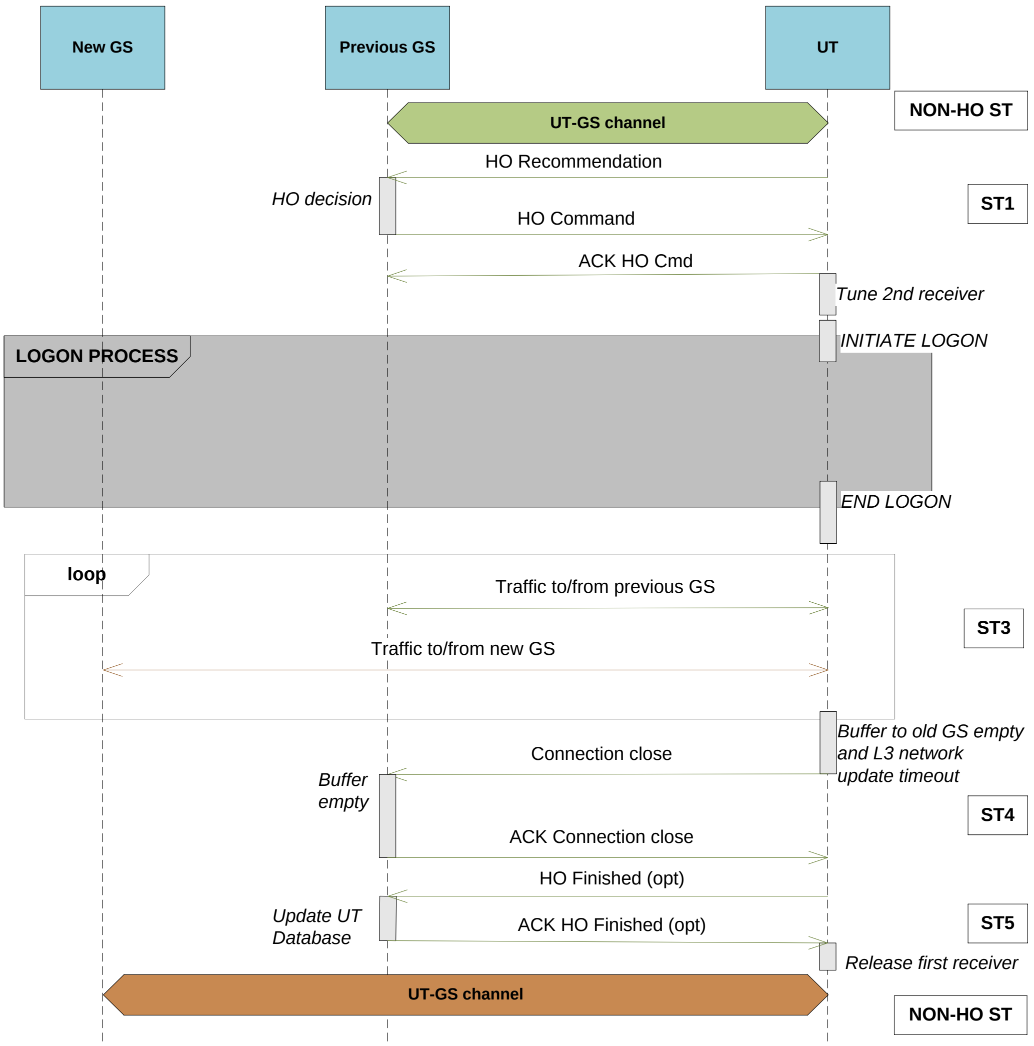
A UT shall be able to handle two independent ACM sessions, one for each of the channels, during HO execution.

9.2.3.3 Handover procedure specification

9.2.3.3.1 "SSP Change" procedure

D018-COM-FUN-2760

The following procedure shall be followed by the UT and GSE for the Satellite Service Provider change handover process and for any handover which involves a change in the NCC element.



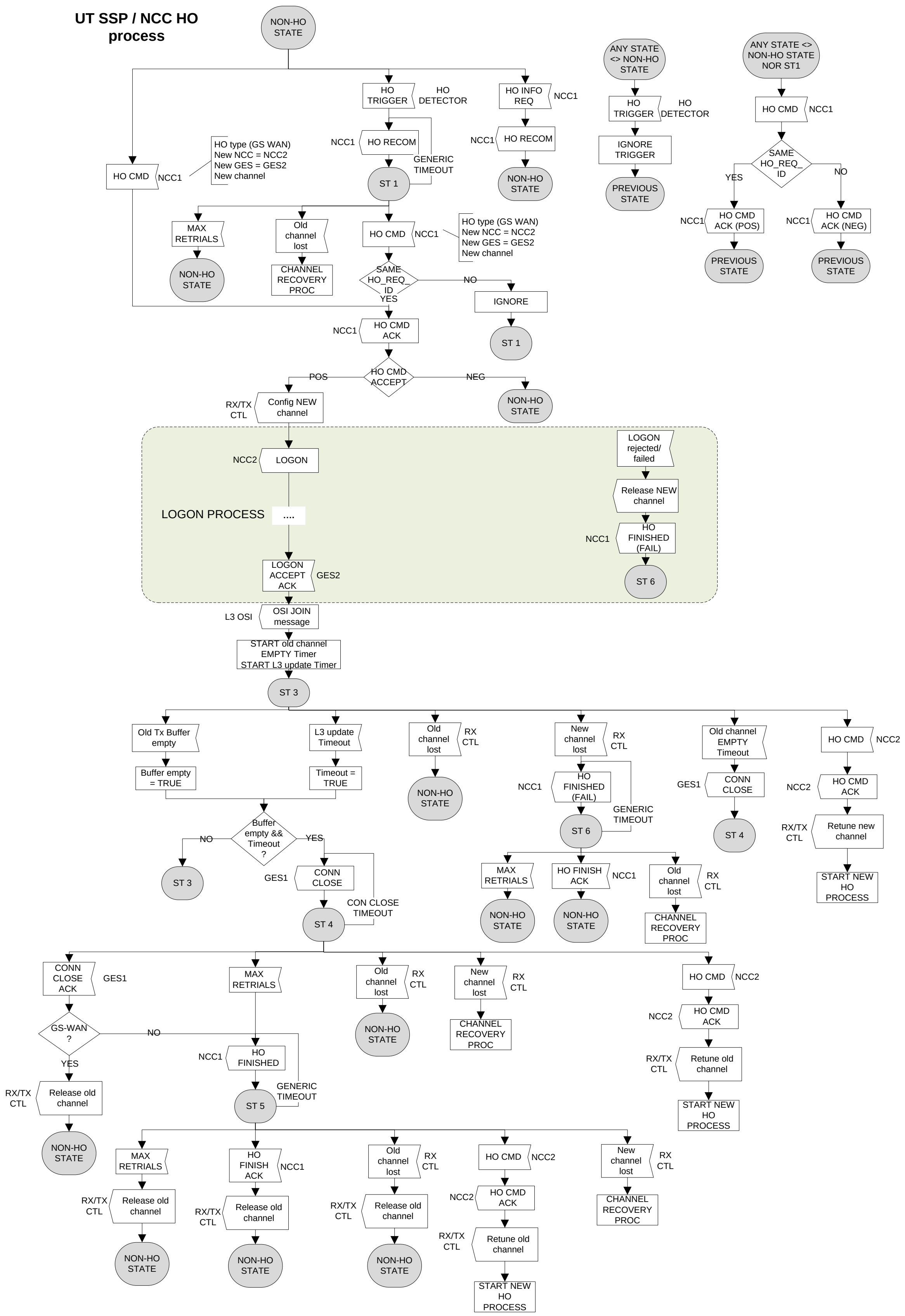
The procedure steps are detailed here:

1. The current GS receives a HO Recommendation message from the UT or GS.
2. The GS validates and decides the target for the handover, and sends HO Command to the UT.
3. The UT confirms with ACK HO Cmd message.
4. The UT tunes its secondary receiver to the new channel.
5. The UT performs LOGON process with the new GS.
6. The UT starts transmitting in parallel to both GSs:
    a. It continues sending fragments of pending L3 packets and new L3 packets with the previous GS destination address to the previous GES through the old channel.
    b. It starts sending incoming L3 packets with the new GS destination address to the new GES through the new channel.
7. Both GSs transmit traffic to the UT through their respective channels.

8. Once the UT empties its buffer of packets to be sent through previous channel and L3 network update timeout has expired, the UT sends Connection Close message to previous GS.
9. Once the previous GS empties its buffer of packets to the UT, it sends ACK Connection Close message to the UT.
10. The UT sends HO Finished message to the previous GS, which responds with ACK HO finished message and updates information related to the UT in its database. This step is optional depending on GS architecture and will be indicated by the GS as D018-COM-FUN-2702.
11. The UT releases first receiver, which was devoted to previous carrier.

D018-COM-FUN-2761

The following SDL shall be followed by the UT for the Satellite Service Provider change handover process and for any handover which involves a change in the NCC element.



UT SSP / NCC HO process
NON-HO STATE
HO TRIGGER
HO DETECTOR
HO INFO REQ
NCC1
HO RECOM
GENERIC TIMEOUT
ST 1
HO CMD
HO type (GS WAN)
New NCC = NCC2
New GES = GES2
New channel
MAX RETRIALS
Old channel lost
CHANNEL RECOVERY PROC
SAME HO_REQ_ID
YES
NO
IGNORE
HO CMD ACK
HO CMD ACCEPT
POS
NEG
RX/TX CTL
Config NEW channel
ANY STATE <> NON-HO STATE
IGNORE TRIGGER
PREVIOUS STATE
ANY STATE <> NON-HO STATE NOR ST1
HO CMD ACK (POS)
HO CMD ACK (NEG)
LOGON PROCESS
NCC2
LOGON
....
LOGON ACCEPT ACK
GES2
LOGON rejected/ failed
Release NEW channel
HO FINISHED (FAIL)
ST 6
L3 OSI
OSI JOIN message
START old channel EMPTY Timer START L3 update Timer
ST 3
Old Tx Buffer empty
Buffer empty = TRUE
L3 update Timeout
Timeout = TRUE
Buffer empty && Timeout ?
GES1
CONN CLOSE
CON CLOSE TIMEOUT
ST 4
RX CTL
New channel lost
Old channel EMPTY Timeout
Retune new channel
START NEW HO PROCESS
HO FINISH ACK
CONN CLOSE ACK
GS-WAN ?
Release old channel
HO FINISHED
ST 5
Retune old channel

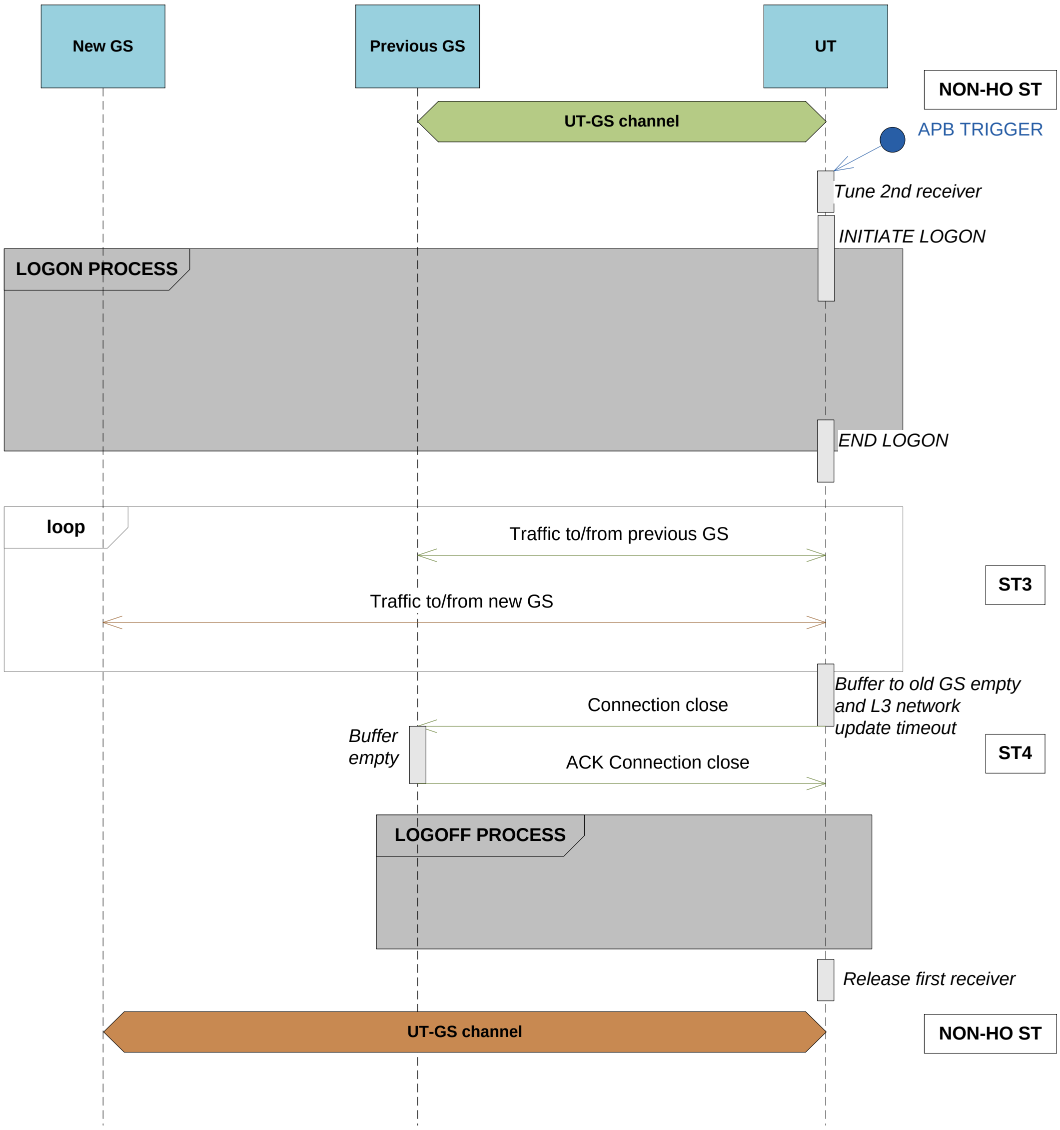
Notes:

- *NON-HO STATE* represents any state in which a HO process is not currently being performed.
- *ST1* corresponds to the state in which the UT is waiting for the HO COMMAND message from GS.
- The type of HO is received by the UT in the HO COMMAND message.
- *LOGON PROCESS* is the process followed to register to a new SSP or NCC.
- *ST3* corresponds to the state in which the UT is transmitting in parallel through old and new channels waiting to empty old transmission buffer.
- *ST4* corresponds to the state in which the UT is waiting for the CONNECTION CLOSE ACK message from GS.
- *ST5* corresponds to the state in which the UT is waiting for the HO FINISH ACK message from the GS after successfully performing the HO.
- *ST6* corresponds to the state in which the UT is waiting for the HO FINISH ACK message from the GS after a failure in the HO process.
- *CHANNEL RECOVERY PROCESS* represents the process to be performed by the UT when the FLC channel is lost.
- *START NEW HO PROCESS* represents the restart of the process with the new received parameters, as if the HO COMMAND message was received in the NON-HO state.
- *GENERIC TIMEOUT* is the timeout defined for signalling messages response from GS.
- *L3 UPDATE TIMEOUT* is the timeout defined to allow the L3 process to update addresses accordingly to the new GS (in the event it is being executed while within the L2 HO process).
- *CONN_CLOSE TIMEOUT* is the timeout defined while waiting for the CONNECTION CLOSE ACK message from the GS
- *OLD CHANNEL EMPTY TIMER* is the timeout defined to complete the whole process of L3 update and emptying old buffer; if this maximum value is exceeded, the old channel will be released anyway.
- *MAX RETRIALS* is the maximum number of retrials for a message retransmission.

9.2.3.3.2 "Direct LOGON in SSP change" procedure

D018-COM-FUN-2770

The following procedure shall be followed by the UT and GSE for the Direct LOGON in new SSP handover process.



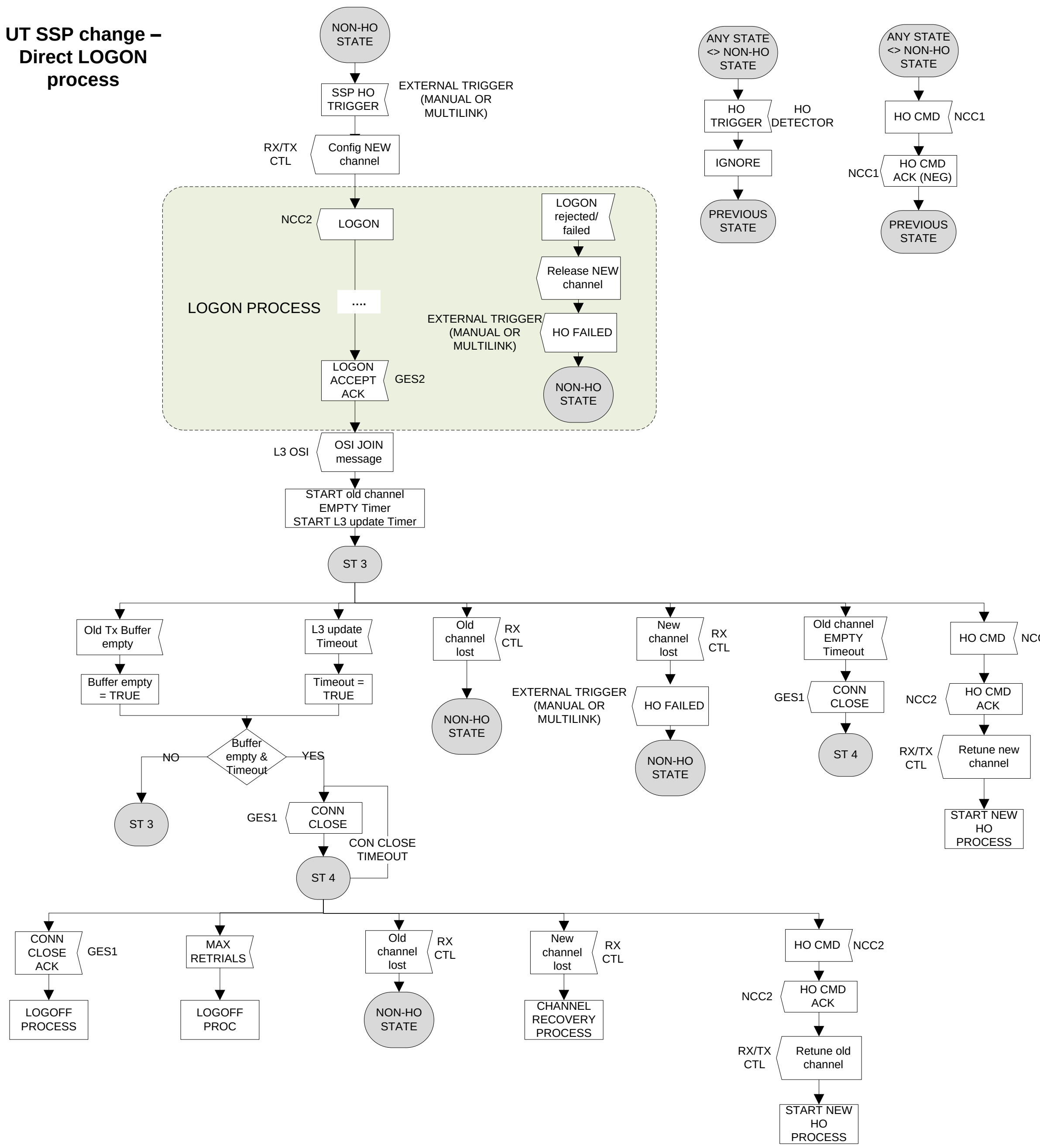
The procedure steps are detailed here:

1. The UT receives a trigger to perform Direct LOGON to a new SSP (manual command from aircrew or APB reasons).
2. The UT tunes its secondary receiver to the new channel and performs LOGON process on the new SSP.
3. The UT starts transmitting in parallel to both GSs:
    a. It continues sending fragments of pending L3 packets to the previous GS through old channel and new L3 packets with the previous GS destination address.
    b. It starts sending incoming L3 packets with the new GS destination address to the new GES through new channel.
4. Both GSs transmit traffic to the UT through their respective channels.

5. Once the UT empties its buffer of packets to be sent through previous channel and L3 network update timeout has expired, the UT sends Connection Close message to the previous GS.
6. Once the previous GS empties its buffer of packets to the UT, it sends ACK Connection Close message to the UT.
7. The UT performs LOGOFF process in the previous GS.
8. The UT releases first receiver, which was devoted to previous carrier.

D018-COM-FUN-2771

The following SDL shall be followed by the UT for the Direct LOGON in new SSP handover process:



Notes:

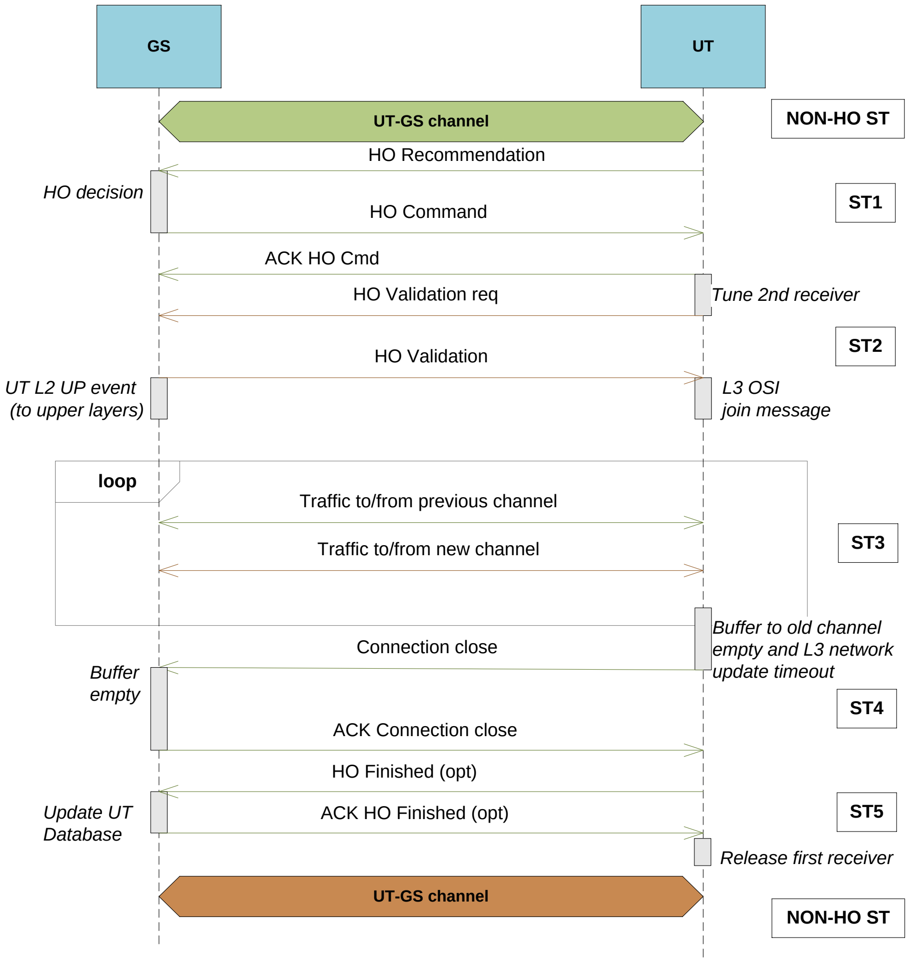
- *NON-HO STATE* represents any state in which a HO process is not currently being performed.
- *LOGON PROCESS* is the process followed to register to a new SSP.
- *ST3* corresponds to the state in which the UT is transmitting in parallel through old and new channels waiting to empty old transmission buffer.
- *ST4* corresponds to the state in which the UT is waiting for CONNECTION CLOSE ACK message from the GS.
- *LOGOFF PROCESS* is the process followed to logoff from a SSP.

- *CHANNEL RECOVERY PROCESS* represents the process to be performed by the UT when the FLC channel is lost.
- *START NEW HO PROCESS* represents the restart of the process with the new received parameters, as if the HO COMMAND message was received in the NON-HO state.
- *GENERIC TIMEOUT* is the timeout defined for signalling messages response from the GS.
- *L3 UPDATE TIMEOUT* is the timeout defined to allow the L3 process to update addresses accordingly to the new GS (in the event it is being executed while within the L2 HO process).
- *CONN_CLOSE TIMEOUT* is the timeout defined while waiting for the CONNECTION CLOSE ACK message from the GS.
- *OLD CHANNEL EMPTY TIMER* is the timeout defined to complete the whole process of L3 update and emptying old buffer; if this maximum value is exceeded, the old channel will be released anyway.
- *MAX RETRIALS* is the maximum number of retrials for a message retransmission.

9.2.3.3.3 "Beam/channel/satellite change within same GES and GES HO" procedure

D018-COM-FUN-2780

The following procedure shall be followed by the UT and GSE for the cases of beam, channel or satellite change within the same GSE and GES change.



The procedure steps are detailed here:

1. Current GS receives a HO Recommendation message from the UT or GS.
2. The GS validates and decides the target for the handover, and sends HO Command to the UT.
3. The UT confirms with ACK HO Cmd message.
4. The UT tunes its secondary receiver to the new channel and transmits HO Validation req message to the GS, which confirms with HO Validation.
5. The UT starts transmitting in parallel in both channels:
    a. It continues sending fragments of pending L3 packets and new L3 packets with the previous GES destination address (only if there is a change of GES) to previous channel.
    b. It starts sending new incoming L3 packets to new channel.
6. Both GS units transmit traffic to the UT through their respective channels.
7. Once the UT empties its buffer of packets to be sent through previous channel, the UT sends Connection Close message to the GS.

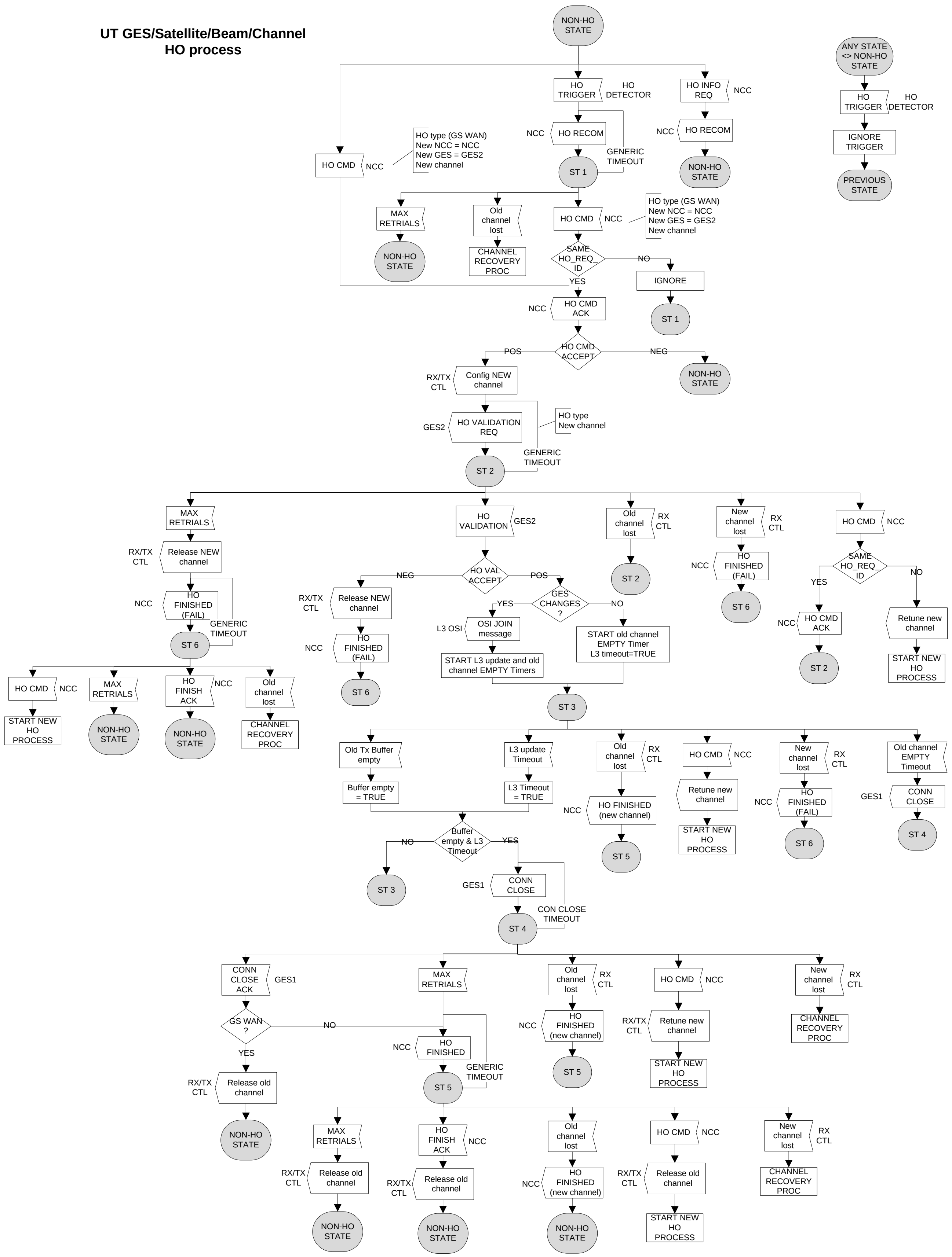
8. Once the GS empties its buffer of packets to the UT, the GS sends ACK Connection Close message to the UT.
9. The UT sends HO Finished message to the GS, which responds with ACK HO finished message and updates information related to the UT in its database. This step is optional depending on GS architecture and will be indicated by the GS as described in D018-COM-FUN-2702.
10. The UT releases first receiver, which was devoted to previous carrier.

Note: Once L2 is correctly established with the UT (L2 UP event), the GS can trigger any associated L3 update processes.

D018-COM-FUN-2781

The following SDL shall be followed by the UT for the cases of beam, channel or satellite change within the same GSE and GES change:

UT GES/Satellite/Beam/Channel HO process



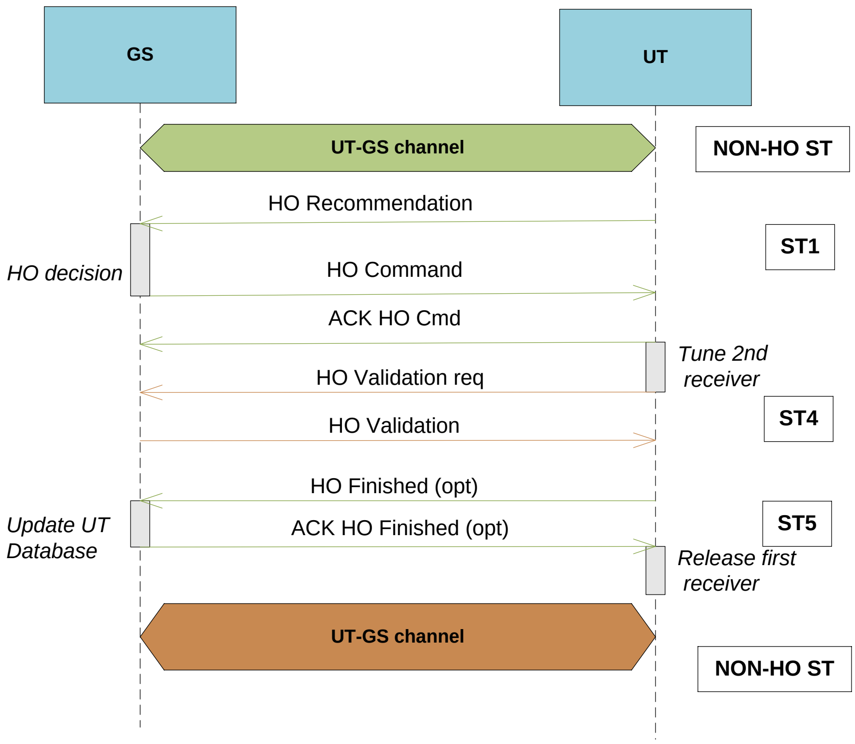
Notes:

- *NON-HO STATE* represents any state in which a HO process is not currently being performed.
- *ST1* corresponds to the state in which the UT is waiting for HO COMMAND message from the GS.
- The type of HO is received by the UT in the HO COMMAND message
- *ST2* corresponds to the state in which the UT is waiting for HO VALIDATION message from the GS.
- *ST3* corresponds to the state in which the UT is transmitting in parallel through old and new channels waiting to empty old transmission buffer.
- *ST4* corresponds to the state in which the UT is waiting for CONNECTION CLOSE ACK message from the GS.
- *ST5* corresponds to the state in which the UT is waiting for HO FINISH ACK message from the GS after successfully performing the HO.
- *ST6* corresponds to the state in which the UT is waiting for HO FINISH ACK message from the GS after a failure in the HO process.
- *CHANNEL RECOVERY PROCESS* represents the process to be performed by the UT when the FLC channel is lost.
- *START NEW HO PROCESS* represents the restart of the process with the new received parameters, as if the HO COMMAND message was received in the NON-HO state
- *GENERIC TIMEOUT* is the timeout defined for signalling messages response from the GS.
- *L3 UPDATE TIMEOUT* is the timeout defined to allow the L3 process to update addresses accordingly to the new GS (in the event it is being executed while within the L2 HO process).
- *CON_CLOSE TIMEOUT* is the timeout defined while waiting for the CONNECTION CLOSE ACK message from the GS.
- *OLD CHANNEL EMPTY TIMER* is the timeout defined to complete the whole process of L3· update and emptying old buffer; if this maximum value is exceeded, the old channel will be released anyway.
- *MAX RETRIALS* is the maximum number of retrials for a message retransmission.

9.2.3.3.4 "FAST handover" procedure

D018-COM-FUN-2790

The following procedure shall be followed by the UT and GSE for the cases of beam, channel or satellite change within the same GSE only in the event that L2 encapsulation and ARQ is performed by a single process common to all Tx/Rx units in the GSE:



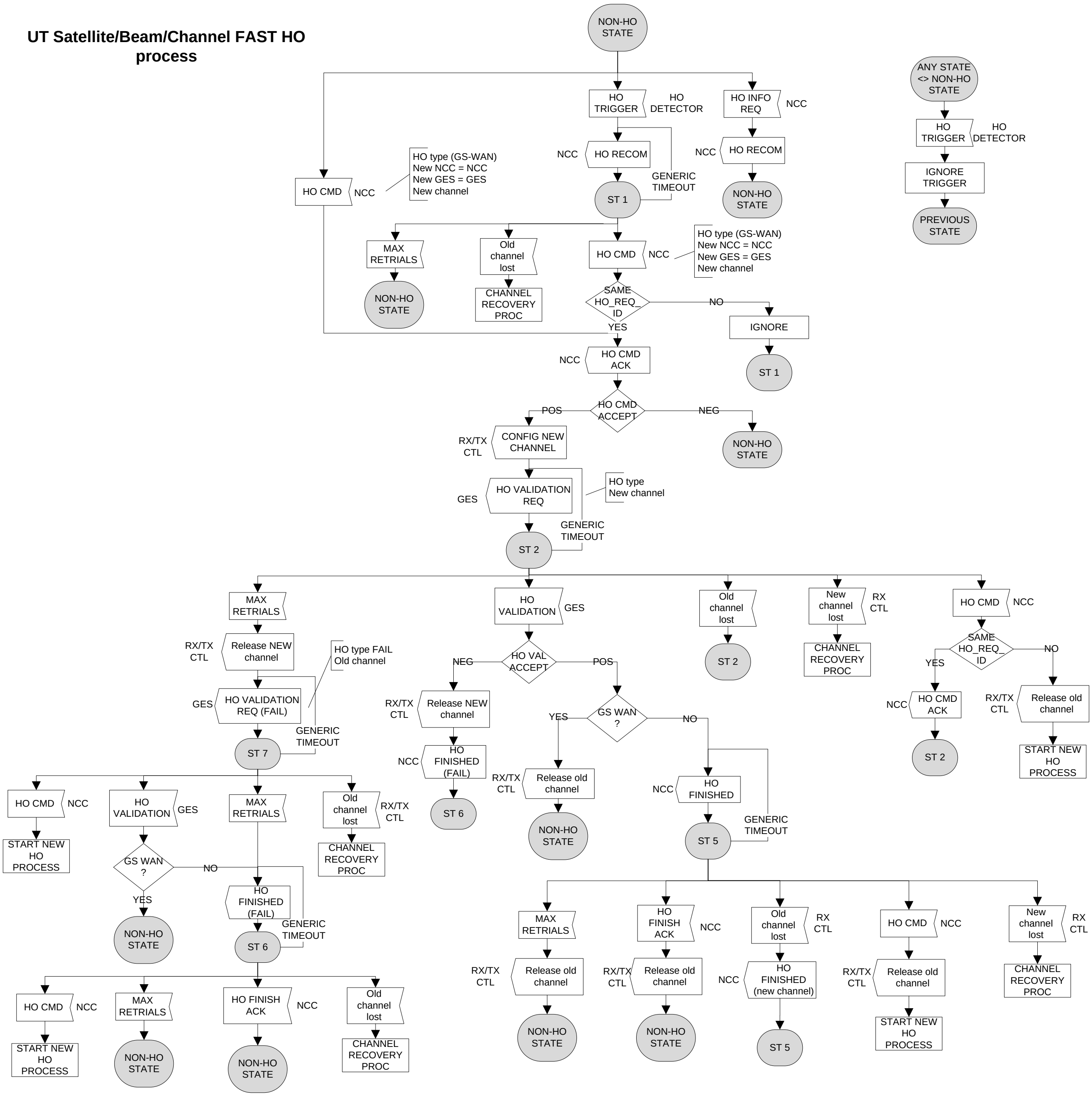
The procedure steps are detailed hereafter:

1. The GS a HO Recommendation message from the UT or GS.
2. The GS validates and decides the target for the handover, and sends HO Command to the UT.
3. The UT confirms with ACK HO Cmd message.
4. The UT tunes its secondary receiver to the new channel and sends HO Validation req message to the GS, which responds with HO Validation.
5. The UT starts sending all traffic through new link.
6. The UT sends HO Finished message to the GS, which responds with ACK HO finished message and updates information related to the UT in its database. This step is optional depending on GS architecture and will be indicated by the GS as described in D018-COM-FUN-2702.
7. The UT releases the first receiver, which was devoted to previous carrier.

D018-COM-FUN-2791

The following SDL shall be followed by the UT for the cases of beam, channel or satellite change within the same GSE only in the event that L2 encapsulation and ARQ is performed by a single process common to all Tx/Rx units in the GSE.

UT Satellite/Beam/Channel FAST HO process



Notes:

- *NON-HO STATE* represents any state in which a HO process is not currently being performed.
- *ST1* corresponds to the state in which the UT is waiting for HO COMMAND message from the GS.
- The type of HO is received by the UT in the HO COMMAND message.
- *ST2* corresponds to the state in which the UT is waiting for HO VALIDATION message from the GS.
- *ST5* corresponds to the state in which the UT is waiting for HO FINISH ACK message from the GS after successfully performing the HO.

- *ST6* corresponds to the state in which the UT is waiting for HO FINISH ACK message from the GS after a failure in the HO process.
- *ST7* corresponds to the state in which the UT is waiting for HO VALIDATION message from the GS after a failure in the HO process.
- *CHANNEL RECOVERY PROCESS* represents the process to be performed by the UTwhen the FLC channel is lost.
- *START NEW HO PROCESS* represents the restart of the process with the new received parameters, as if the HO COMMAND message was received in the NON-HO state.
- *GENERIC TIMEOUT* is the timeout defined for signalling messages response from GS.
- *MAX RETRIALS* is the maximum number of retrials for a message retransmission.

9.2.3.3.5 "Bulk Handover" procedure for non-GEO satellites

Note: the concept of "Bulk Handover" is defined as a HO in which all traffic in a beam or in a complete satellite is handed over to another satellite in a single operation involving several UTs.

D018-COM-FUN-2810

The GS shall be able to pre-synchronise to the new satellite prior to performing the actual switch of all traffic from the old satellite to the new one.

Note: it is assumed that the time interval in which the target service area is fully covered by both ascending and descending non-GEO satellites is known by the GS.

D018-COM-FUN-2820

The GS shall switch to the new satellite once the synchronisation with the new satellite is achieved, and stop all transmission to the old satellite.

D018-COM-FUN-2821

The GS shall be able to toggle the DPC_RST and ACM_RST bits of the FWD_DD when switching to the new satellite.

Note: D018-COM-FUN-3211 and D018-COM-FUN-3351 describe actions performed by the UT.

9.2.4 Network synchronisation procedures

In this section the following concepts are used:

- Feeder-to-Feeder links: GS-Satellite-GS fixed links operating at Ku, Ka or C band.
- Satellite ATM transceiver: satellite transceivers from/to Ku, Ka or C band to/from L band.

9.2.4.1 General synchronisation aspects

The requirements presented in this section affect both forward and return link network synchronisation.

D018-COM-FUN-2920

All GS elements shall compensate the Feeder link Doppler and Doppler rate with a normalised residual error lower than 10^{-3} by:

- Computing the Doppler Effect from
  - GS element location
  - Satellite location and speed derived from satellite ephemerides
  - Nominal carrier and symbol/chip frequencies
- Compensating Feeder link Doppler Effect on
  - Transmitter frequencies and time
  - Receiver frequencies and time

Feeder Link Doppler Effect compensation is mandatory for HEO and MEO constellations and optional for GEO constellations.

9.2.4.2 GS network synchronisation procedures

9.2.4.2.1 NCC network synchronisation procedures

D018-COM-FUN-2950

The NCC shall implement the following network synchronisation procedures:

- Feeder link Doppler pre-compensation in both uplink and downlink.
- Satellite clock error* estimation and distribution to all GS elements.
- Satellite translation error compensation in both forward link transmissions and return link receptions.
- Network clock reference (NCR) distribution to all GS elements.
- Reception of its own transmissions.
- SNIR estimation.
- Application of power correction to compensate for uplink fades.

* *The satellite clock error is the satellite clock frequency bias, usually expressed in ppm.*

9.2.4.2.2 GES network synchronisation procedures

D018-COM-FUN-2960

The GES shall implement the following network synchronisation procedures:

- Feeder link Doppler pre-compensation in both uplink and downlink.
- Reception of the NCC and the own GES transmissions.
- NCR counter reception (from FWD_DD NCR field) and Network clock reference recovery.
- Satellite clock error reception.
- Satellite translation error compensation in both forward link transmissions and return link receptions.

D018-COM-FUN-2970

The GES, at start up, shall implement the following forward link initial synchronisation procedure:

- Execution of the processes defined in D018-COM-FUN-2960.
- Round trip delay estimation from the satellite and its own locations.
- Computation of the initial transmission power based on link budgets. Alternatively, the initial transmission power can be provided to the GES as a configuration parameter.
- Transmission to a timeslot assigned to the GES. The recovered time reference (NCR) is used and the estimated round trip delay compensated.
- Reception of its own transmission.
- Estimation of the synchronisation (frequency and time) and SNIR errors with respect to those transmissions from the NCC.
- Synchronisation (frequency and time) and application of power corrections (to compensate for uplink fades).

After completing the previous steps, the GES starts the forward link synchronisation maintenance procedure.

D018-COM-FUN-2980

The GES shall implement the following forward link synchronisation maintenance procedure:

- Execution of the processes defined in D018-COM-FUN-2960.
- Transmission to timeslots assigned to the GES. The recovered time reference (NCR) is used.
- Reception of its own transmissions.
- Estimation of the synchronisation (frequency and time) and SNIR errors with respect to those transmissions from the NCC.

- Synchronisation (frequency and time) and application of power corrections.

D018-COM-FUN-2990

The GES shall interrupt transmissions and go back to the forward link initial synchronisation stage, upon any of the following events:

- When no transmissions from NCC are received for NCC_loss_timeout seconds.
- When GES_n_loss consecutive GES transmissions are not received.
- When the Network clock reference is not properly recovered.

9.2.4.3 Forward link synchonisation

D018-COM-FUN-3010

Forward link initial synchronisation and synchronisation maintenance procedures shall be implemented

- either by means of Feeder-to-Feeder links
- or through the Forward Link Carrier.

D018-COM-FUN-3020

The NCC shall distribute to all GS elements the Network Clock Reference (NCR)

- either by broadcasting NCR counter through the FCH physical channel (FWD_DD NCR field), if the forward link network synchronisation is implemented through the Forward Link Carrier;
- or by broadcasting NCR counter through specific channels, if the forward link network synchronisation is implemented through Feeder-to-Feeder links.

D018-COM-FUN-3030

The Network Clock Reference shall have a frequency of 27 MHz.

D018-COM-FUN-3040

The FWD_DD NCR field shall contain the value of a 40-bit counter provided in tics of the Network Clock Reference. The value corresponds

- either to the time at which the last preamble symbol of the FCH burst containing the NCR counter is transmitted, if the forward link network synchronisation is implemented through the Forward Link Carrier
- or to the time at which the last preamble symbol of the specific channel burst containing the NCR counter is transmitted, if the forward link network synchronisation is implemented through Feeder-to-Feeder links.

D018-COM-FUN-3050

The NCR counter is inserted with a periodicity that guarantees a proper recovery of the NCR to be used as a system time reference by any GS element.

- The minimum NCR counter periodicity shall be 0.5 packet/s.

D018-COM-FUN-3060

GS elements shall recover the Network Clock Reference from the received NCR counter (FWD_DD NCR field).

D018-COM-FUN-3110

The forward link initial network synchronisation procedure shall guarantee that GS transmissions to the Forward Link Carrier at the satellite front-end are

- Synchronised in time: maximum time error lower than half the guard time.
- Synchronised in carrier frequency: maximum frequency error lower than half the guard band.

D018-COM-FUN-3120

The forward link network synchronisation maintenance procedure shall guarantee that GS transmissions to the Forward Link Carrier at the satellite front-end are

- Synchronised in time: maximum time error lower than half the guard time.
- Synchronised in carrier frequency: maximum frequency error lower than 10 Hz.
- Power balanced: uplink fades are compensated.

D018-COM-FUN-3130

The SYNC* field of the FWD_DD descriptor of FCH bursts shall be filled as follows:

- 1 for all FCH bursts transmitted by the NCC.
- 1 for those FCH bursts transmitted by a GES while it is in the forward link network synchronisation maintenance stage.
- 0 otherwise.

* *Note that the SYNC field is actually used by the UT to implement the transmitter Doppler pre-compensation in the return uplink. However, its value depends on the forward link network synchronisation stage of the GS element transmitting the FCH burst.*

9.2.4.4 Return Link synchronisation

The return link synchronisation procedures are depicted in Figure 9-1 and covered in the requirements in subsequent sections.

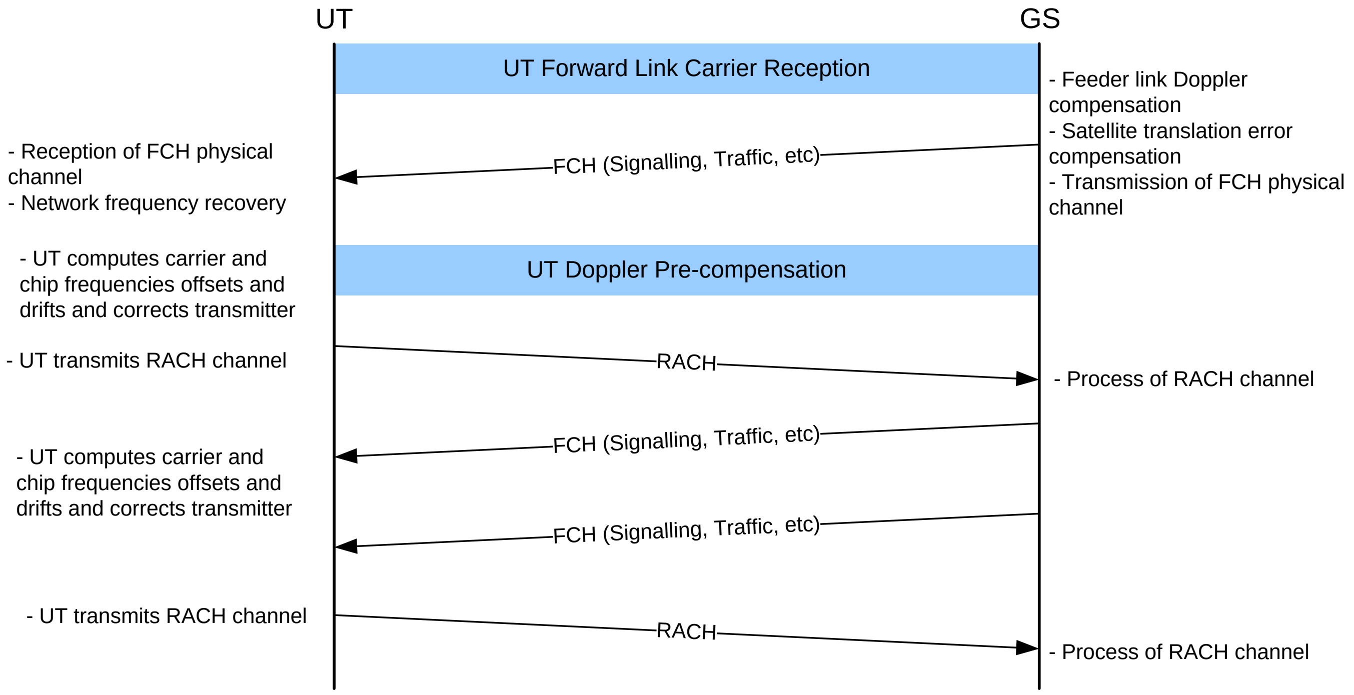


Figure 9-1: Return link synchronisation procedure

9.2.4.4.1 UT Forward Link Carrier reception

D018-COM-FUN-3160

The UT shall be able to detect, demodulate and decode Forward Link Carrier bursts affected by significant carrier frequency offsets. The sources of carrier frequency error are:

- GS (NCC or GES) reference clock instability
- Satellite reference clock instability (residual error after the compensation implemented by GS elements)
- UT reference clock instability
- Satellite motion (residual error after the Feeder link Doppler compensation implemented by GS elements)
- UT motion

The maximum carrier frequency offsets are:

- 7 kHz for GEO constellations (Feeder link operation in Ka-band is assumed)

- 22.5 kHz for MEO constellations (Feeder link operation in C-band is assumed)
- 37 kHz for HEO constellations (Feeder link operation in Ka-band is assumed)

D018-COM-FUN-3170

The UT shall be able to detect, demodulate and decode Forward Link Carrier bursts affected by significant time drifts. The sources of time drifts are:

- GS (NCC or GES) reference clock instability
- UT reference clock instability
- Satellite motion (residual error after the Feeder link Doppler compensation implemented by GS elements)
- UT motion

The maximum time drifts are:

- 3.9 us/s for GEO constellations
- 14.2 us/s for MEO constellations
- 22.6 us/s for HEO constellations

D018-COM-FUN-3180

The UT shall be able to detect, demodulate and decode Forward Link Carrier bursts affected by significant carrier frequency drifts. The sources of carrier frequency drifts are:

- Satellite acceleration (residual error after the Feeder link Doppler compensation implemented by GS elements)
- UT acceleration and angular movement

The maximum carrier frequency drift is 350 Hz/s.

D018-COM-FUN-3190

The UT shall be able to detect, demodulate and decode Forward Link Carrier bursts affected by significant time drift variations. The sources of time drift variations are:

- Satellite acceleration (residual error after the Feeder link Doppler compensation implemented by GS elements)
- UT acceleration and angular movement

The maximum time drift variation is 0.22 us/s^2.

9.2.4.4.2 UT Doppler pre-compensation and Power Randomization

D018-COM-FUN-3200

The UT shall implement a transmitter Doppler pre-compensation mechanism aimed at:

- Minimising the carrier and chip frequency errors affecting RACH bursts
- Estimating and compensating for the return uplink Doppler dynamics to reduce synchronisation errors

D018-COM-FUN-3210

The UT transmitter Doppler pre-compensation mechanism shall be based on the receiver carrier frequency offsets estimated by the UT demodulator upon reception of the Forward Link Carrier.

D018-COM-FUN-3211

The UT shall reset the UT Doppler pre-compensation mechanism every time the DPC_RST bit of the FWD_DD changes its state (see requirement D018-COM-FUN-1031 above).

D018-COM-FUN-3220

The UT shall discard the following FCH bursts to implement the transmitter Doppler pre-compensation mechanism:

- FCH bursts recovered with errors (wrong CRC).
- FCH bursts with the SYNC field of the FWD_DD descriptor set to 0.

D018-COM-FUN-3230

The UT transmitter Doppler pre-compensation mechanism shall estimate the return uplink Doppler shift and drift by:

- Estimating the forward downlink Doppler shift and drift from the carrier frequency offsets estimated by the reception of FCH bursts.
- Deriving the return uplink Doppler shift and drift from the above estimations.

D018-COM-FUN-3240

The UT shall adjust its transmitter carrier and chip frequencies according to the estimated return uplink Doppler shift and drift.

D018-COM-FUN-3250

A UT, during the satellite HO execution process, shall be capable of transmitting not simultaneously two different sets of RTN link A-CDMA carriers, applying the corresponding transmitter Doppler pre-compensation for each carrier on a burst-by-burst basis.

D018-COM-FUN-3260

The UT shall guarantee that RACH bursts are received at the GS in the target contention channel with a carrier frequency error lower than:

- 4.5 kHz for GEO constellations (Feeder link operation in Ka-band is assumed).
- 3.8 kHz for MEO constellations (Feeder link operation in C-band is assumed).
- 5.8 kHz for HEO constellations (Feeder link operation in Ka-band is assumed).

D018-COM-FUN-3270

The UT shall guarantee that RACH bursts, once received at the GS, are affected by a maximum carrier frequency drift of 50 Hz/s.

D018-COM-FUN-3280

The UT shall guarantee that RACH bursts, once received at the GS, are affected by a time drift during the burst duration lower than $T_c/8$, T_c being the chip period.

D018-COM-FUN-3290

The UT shall interrupt transmissions in any of the following circumstances:

- When required to do so by the GS.
- When the Forward Link Carrier is lost, i.e., no FCH bursts are received for FLC_loss_timeout seconds.
- When the UT transmitter Doppler pre-compensation mechanism is not able to estimate the return uplink Doppler shift and drift.

D018-COM-FUN-3295

The UT shall randomize the power of the transmission in a per burst basis using the procedure shown below:

- The UT identifies the value of the *lower_bound_power_dB* parameter associated with the channel in which the burst is transmitted. This parameter is distributed by the GS through System Tables.
- A random power factor *k_dB* is computed following a uniform distribution (in the logarithmic domain) in the range [*lower_bound_power_dB* , 0].
- The effective burst's transmission power is computed as *EIRP_dB* + *k_dB*.

Note: the value of the parameter lower_bound_power_dB is 0 or a negative value.

9.2.4.4.3 GS return link network synchronisation

The return link network synchronisation procedures implemented by the GS elements are common to the forward and return links and are included in section 9.2.5.2, the most relevant being the:

- Feeder link Doppler pre-compensation in both uplink and downlink.
- Satellite translation error compensation in both forward link transmissions and return link receptions.

D018-COM-FUN-3300

GS elements shall be able to detect, demodulate and decode RACH bursts affected by a carrier frequency error of up to:

- 4.5 kHz for GEO constellations (Feeder link operation in Ka-band is assumed).
- 3.8 kKz for MEO constellations (Feeder link operation in C-band is assumed).
- 5.8 kHz for HEO constellations (Feeder link operation in Ka-band is assumed).

D018-COM-FUN-3310

GS elements shall be able to detect, demodulate and decode RACH bursts affected by a carrier frequency drift of up to 50 Hz/s.

D018-COM-FUN-3320

GS elements shall be able to detect, demodulate and decode RACH bursts affected by a time drift during the burst duration of up to $T_c/8$, T_c being the chip period.

9.2.5 ACM

9.2.5.1 General ACM requirements

D018-COM-FUN-3330

ACM shall be supported on the Forward Link user plane only, the supported MODCODs being those specified in requirement D018-COM-FUN-0850 above.

D018-COM-FUN-3340

The UT shall permanently monitor the Link Quality of the received signal in its serving beam, the Link Quality being based on PER measurements.

D018-COM-FUN-3350

Based on Link Quality measurements, the UT ACM module shall be able to determine the least robust FWD link MODCOD the GS can use to communicate with the UT, while keeping the target PER, in order to maximise the spectral efficiency.

Note: in the following requirements, the least robust MODCOD supported by a UT is referred to as "preferred MODCOD".

D018-COM-FUN-3351

The UT shall reset the ACM Link Quality measurements every time the ACM_RST bit of the FWD_DD changes its state (see requirement D018-COM-FUN-1032 above).

D018-COM-FUN-3360

A UT, during a HO execution process, shall be able to maintain two ACM sessions.

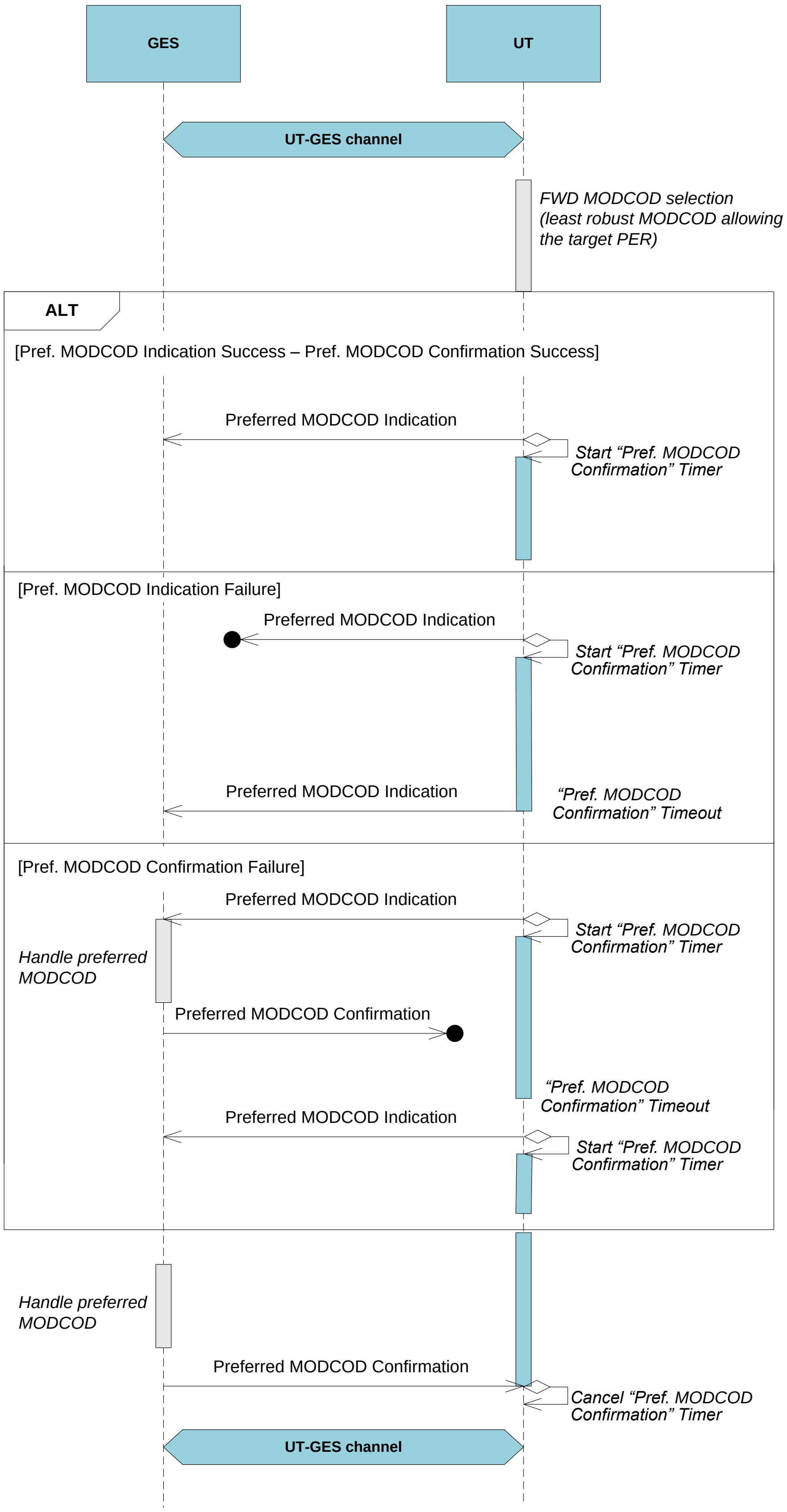
9.2.5.2 ACM procedures

D018-COM-FUN-3370

At the UT power-up, the GS shall use the most robust MODCOD to communicate with the UT.

D018-COM-FUN-3380

In the event that the preferred MODCOD changes, the UT shall indicate the new preferred MODCOD to the GS according to the Preferred MODCOD Indication procedure detailed below.



The following steps are to be performed during the Preferred MODCOD Indication procedure:

(Note: the UT continuously monitors the Link Quality of the received signal in the serving beam.)

(Nominal case: Pref. [MODCOD Indication success - Pref. MODCOD Confirmation success])

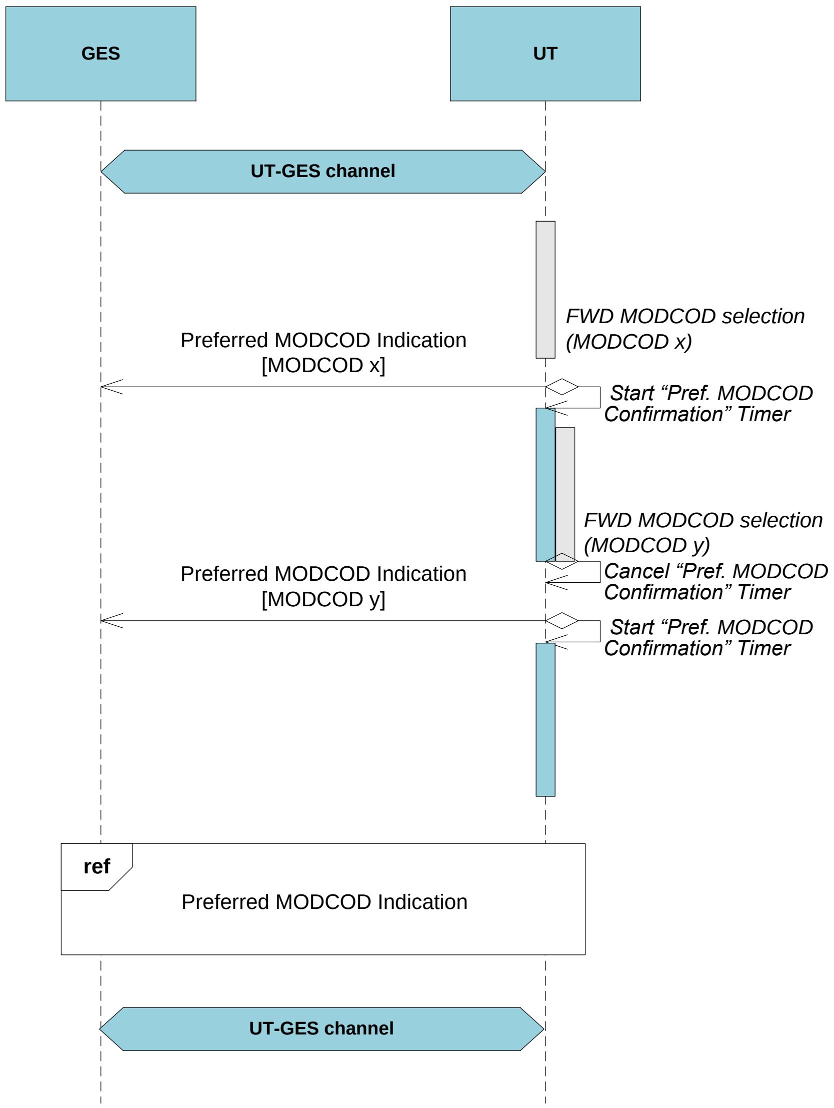
1. Based on Link Quality measurements, the UT determines the most efficient FWD Link MODCOD the GS can use to communicate with it.
    a. If the preferred MODCOD changes, the UT sends Preferred MODCOD Indication message to the GES.
    b. Otherwise, the UT does not send any message and continues monitoring the Link Quality of the received signal.
2. In order to prevent signalling messages losses, "Preferred MODCOD Confirmation" timer is triggered on the UT end.
3. Upon correct reception of the Preferred MODCOD Confirmation message, the GES sends Preferred MODCOD Confirmation message to the UT.
4. On the UT end, after reception of the Preferred MODCOD Confirmation message, the UT cancels the Preferred MODCOD Confirmation timer.

In the event that signalling messages are lost:

- In the event of Preferred MODCOD Indication failure, after expiration of "Preferred MODCOD Confirmation" timer, the UT retransmits the Preferred MODCOD Indication message as in Step 1 of the Preferred MODCOD Indication procedure - Nominal case.
- In the event of Preferred MODCOD Confirmation failure, after expiration of "Preferred MODCOD Confirmation" timer, the UT retransmits the Preferred MODCOD Indication message as in Step 1 of the Preferred MODCOD Indication procedure - Nominal case.

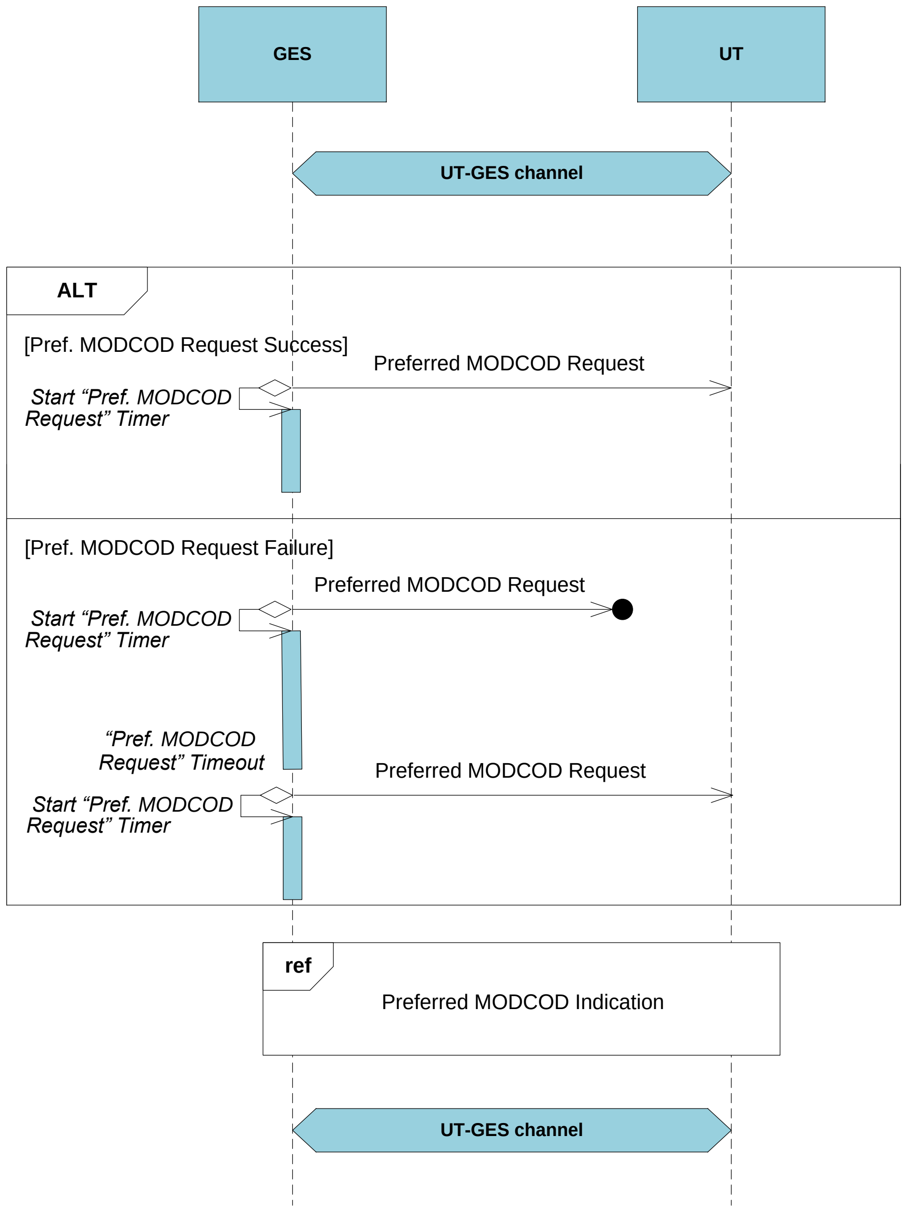
D018-COM-FUN-3390

If the UT detects a MODCOD change before receiving the Preferred MODCOD Confirmation message from the GS, the UT shall send a Preferred MODCOD Indication with the new MODCOD as detailed in the following procedure.



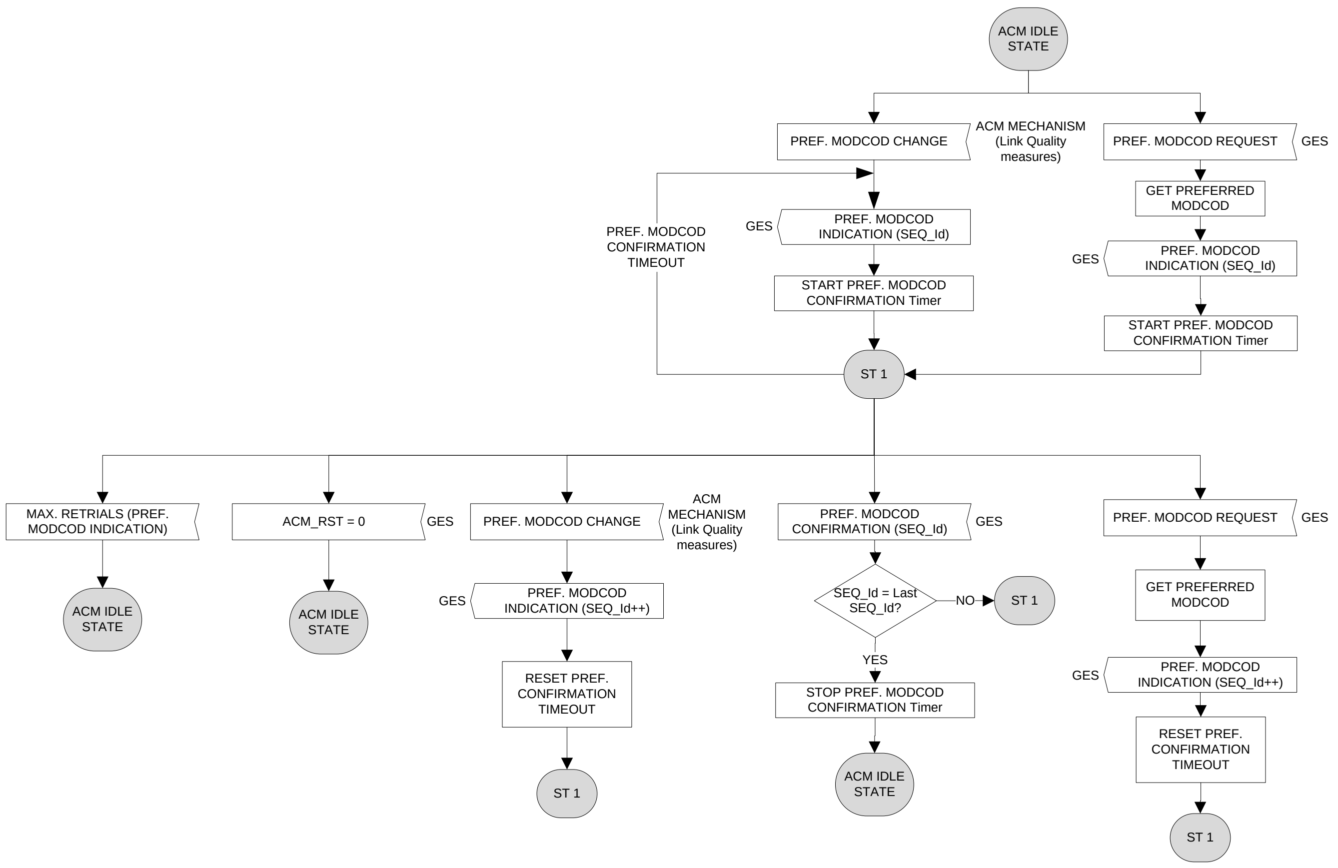
D018-COM-FUN-3400

In the event that the GS needs to request the Preferred MODCOD from a UT, the following procedure shall be followed.



D018-COM-FUN-3401

The following SDL shall be followed by the UT in order to indicate the Preferred MODCOD to the GS.

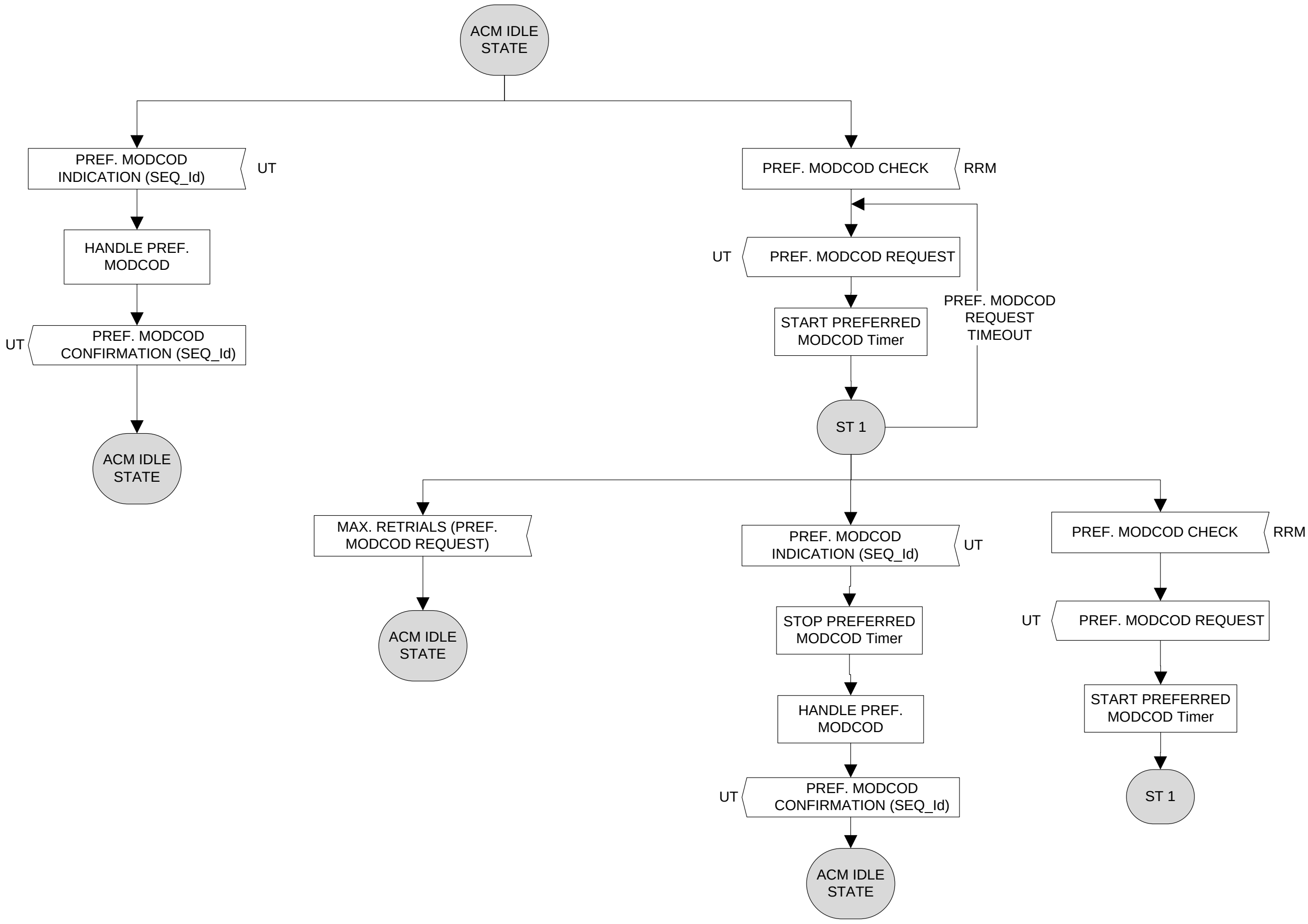


Notes:

- *ACM IDLE STATE* represents the state in which the Preferred MODCOD indication notification has not started.
- *ST1* corresponds to the state in which the UT is waiting for PREFERRED MODCOD CONFIRMATION message from GS.
- *GET PREFERRED MODCOD* represents the process carried out by the UT to get the current preferred MODCOD.
- *PREFERRED MODCOD CONFIRMATION TIMEOUT* is the timeout defined for the signalling messages response from GS.
- *MAX RETRIALS* is the maximum number of retrials for a message retransmission
- *PREFERRED MODCOD CHANGE* represents an event from the ACM mechanism which occurs when a Preferred MODCOD change is detected.
- *SEQ_Id* is the sequence number used to mark each *PREFERRED MODCOD INDICATION* sent to the GS*.*

D018-COM-FUN-3402

The following SDL shall be followed by the GS to manage the Preferred MODCOD of each UT.



Notes:

- *ACM IDLE STATE* represents the state in which the GES is waiting for a PREFERRED MODCOD INDICATION message from the UT or an internal event from RRM which triggers a PREFERRED MODCOD REQUEST message.
- *ST1* corresponds to the state in which the GES is waiting for PREFERRED MODCOD INDICATION message from UT.
- *HANDLE PREFERRED MODCOD* represents the process carried out by the GES to update the preferred MODCOD for the UT which has sent the PREFERRED MODCOD INDICATION message.
- *PREFERRED MODCOD REQUEST TIMEOUT* is the timeout defined for the signalling messages response from UT.
- *MAX RETRIALS* is the maximum number of retrials for a message retransmission.
- *PREFERRED MODCOD CHECK* represents an event from the RRM (or other Rx process) which asks for checking of the current Preferred MODCOD for a given UT.
- *SEQ_Id* is the sequence number used to mark each PREFERRED MODCOD INDICATION received by the GES. The GES replies with a PREFERRED MODCOD CONFIRMATION message marked with the same SEQ_Id.

9.2.6 Radio resource management procedures

This section addresses:

- Congestion of the RACH (data and signalling)

9.2.6.1 Congestion control

The purpose of congestion control (CC) mechanisms is to control the amount of traffic entering in the communications network in order to avoid a collapse caused by oversubscription of either processing abilities or link capabilities of the networks and, furthermore, to ensure network stability, throughput efficiency and a fair allocation of resources.

The meaning of congestion control parameters indicated in this section is defined in section 8.7.1.1 (D018-COM-FUN-1610) and section 8.7.1.2 (D018-COM-FUN-1720) above.

9.2.6.1.1 Return Link congestion control protocol for data channel

D018-COM-FUN-3410

The UT shall support the congestion control mechanism indicated by the FLC through the system signalling messages defined in section 11.5.1:

CC Parameter	CC Status
tx_backoff persistence retransmission_timeout	Traffic status (load)

D018-COM-FUN-3420

The GS shall provide a set of congestion control parameters, as they are defined in D018-COM-FUN-3410 above, according to system load status (low traffic, medium traffic, high traffic, congested) and supported spreading factors.

D018-COM-FUN-3430

The system load status shall be estimated by the GS based on the measured A-CDMA channel noise rise and its evolution, and sent to terminals within the CC_STATUS message.

D018-COM-FUN-3440

The CC_STATUS message shall be sent periodically.

D018-COM-FUN-3450

Terminals shall use the estimated system load status to select the appropriate set of Congestion Control parameters.

D018-COM-FUN-3460

CC_STATUS messages can be sent either as a standalone message or be integrated as a part of the signalling framework.

9.2.7 Support to redundancy and failure detection procedures

Redundancy and failure detection procedures are basically transparent to the CS. Whenever a failure event happens, the communication system will reconfigure itself to use the backup entities, making the UT essentially unaware of the event. However, the CS provides certain support functions, especially with regard to ATN/OSI.

D018-COM-FUN-3463

If the UT receives an OSI RESET message, it shall discard it if the counter RESET_COUNTER is equal to the one of the last valid RESET packet received over the same link.

Note: this requirement is only applicable if the UT has received a previous OSI RESET message over the same link and if it supports OSI.

D018-COM-FUN-3465

If the UT receives a valid OSI RESET message, it shall trigger locally the reset of the 8208 virtual connections associated with the link where this OSI RESET was received.

D018-COM-FUN-3466

Whenever the UT receives a POLL REQUEST signalling message, it shall reply with a POLL RESPONSE message.

9.2.8 Security control procedures

Refer to 8.3.

9.3 Control plane forward link specification

9.3.1 Link layer specification

9.3.1.1 ARQ protocol

Protocols in the control plane provide their own retransmission mechanism. Hence, they do not rely on link layer ARQ procedures.

9.3.1.2 Encapsulation

Refer to User plane (section 8.6.1.1).

9.3.1.3 Security

Refer to 8.3 above.

9.3.2 Physical layer specification

D018-COM-FUN-3470

The forward link control shall use the following burst types:

- FCH burst

D018-COM-FUN-3480

The FWD Link Control Plane shall use the Physical layer specification detailed in section 8.5.2 above for the FWD Link User Plane with the exception that ACM is not supported for signalling purposes.

The signalling bursts shall use QPSK 1/4.

9.4 Control plane return link specification

9.4.1 Link layer specification

9.4.1.1 Random access

D018-COM-FUN-3485

For the transmission over the RTN link of all signalling messages except ARQ ACKs, the UT shall apply the congestion control mechanism described here:

- For the first transmission of a signalling message, it forwards the message directly to the transmission scheduler, bypassing the congestion control.

- For the following transmissions (i.e., for the signalling message retransmissions), the congestion control mechanism defined in steps 2 to 4 in D018-COM-FUN-1610 is followed by the signalling procedure, initializing the value of the variable 'npdu_ntx' to 1 and assuming that the signalling message is the NPDU.

The signalling message uses the congestion control parameters '*tx_backoff*' and '*persistence*' associated with the '*CC_category_id*' with value '*Signalling*'.

Note: ARQ ACK signalling messages are treated as indicated in D018-COM-FUN-1605 and D018-COM-FUN-1610 above.

9.4.1.2 ARQ protocol

Protocols in the control plane provide their own retransmission mechanism. Hence, they do not rely on link layer ARQ procedures.

9.4.1.3 Encapsulation

Refer to section 8.7.1.3 above.

9.4.1.4 Security

Refer to section 8.3 above.

9.4.2 Physical layer specification

D018-COM-FUN-3490

The return link control plane shall use the following burst types:

- RACH burst (Random Access burst)

D018-COM-FUN-3500

The RTN Link Control Plane shall use the Physical layer specification detailed in section 8.7.2 for the RTN Link User Plane.

10. MANAGEMENT PLANE SPECIFICATION

Management of the UT / GS is solely local to the aircraft / ground and does not require any exchange of information over the air interface. Refer also to section 6 above.

11. SIGNALLING STRUCTURES

This section provides detailed definition of the signalling structures to be used for several control procedures.

Message description semantic is the one used in [RD-11].

11.1 System information tables

11.1.1 Logon table

D018-COM-FUN-3520

The LOGON TABLE message shall comply with the following structure:

Syntax	No. of bits
LOGONsystemtable_payload_structure() {	
SigMsgID (0x11)	8
opt_has_FWD_neighbor_infos_flag	1
RTN_channel_definition_per_NCC_flag	1
Reserved	2
NCC_list_number	4
N_neighbors_FWD	4
N_max_CSP_ids	4
//comment: N_CC_cong_states and N_CC_parameter_sets minimum value 1	
N_CC_cong_states	8
N_CC_parameter_sets	4
// comment: N_RACH_burst_configurations minimum value 1	
N_RACH_burst_configurations	4
for (m=0; m < N_RACH_burst_configurations; m++) {	
Number_preamble_scrambling_options[m] // for RACH burst conf	3
}	
if ((3*N_RACH_burst_configurations % 8) !=0) {	
reserved (length= 8-(3*N_RACH_burst_configurations)%8)	2
}	
N_RTN_channels	8
for (n=0; n < N_RTN_channels; n++) {	

N_supported_RACH_confs[n] //Comment: for each RTN Channel	4
}	
if (N_RTN_channels%2 == 1) {	
reserved	4
}	
if (RTN_channel_definition_per_NCC_flag == 1) {	
N_RTN_channels_per_NCC	8
}	
Beam_id	8
System_config_version_number	8
// Comment : General parameters (administrative)	
Administrative_parameters_sys_id	4
Administrative_parameters_SSP_id	4
FWD_link_band_central_frequency	16
RTN_link_band_central_frequency	16
// Comment: RTN link Congestion Control parameters	
// Comment: As a minimum, CC parameter set for signalling	
// Comment: CC_category_id = signalling & CC_SF_id = 16	
for (i=0 ; i < N_CC_parameter_sets; i++) {	
CC_config(N_CC_cong_states)	
}	
// Comment : RTN link burst configuration parameters	
// Comment: As a minimum, config required for logon signalling	
for (n=0; n < N_RACH_burst_configurations; n++) {	
RACH_ burst_configuration(Number_preamble_scrambling_options[n])	
}	
// Comment : RTN link channel configuration parameters	
// Comment: As a minimum, config for one signalling channel	
for (n=0; n < N_RTN_channels; n++) {	
RTN_channel_configuration(N_supported_RACH_confs[n])	

Traffic_status	8
}	
// Comment: Config parameters per NCC	
for (i=0; i < NCC_list_number; i++) {	
Administrative_parameters_NCC_id	4
NCC_ID	8
Logon_request_retransmission_number	4
for (j=0; j < N_max_CSP_ids; j++) {	
Administrative_parameters_CSP_id	4
}	
// Comment : A NCC may use a subset of RTN channels	
if (RTN_channel_definition_per_NCC_flag == 1) {	
for (k=0; k < N_RTN_channels_per_NCC; k++) {	
RTN_channel_id	8
}	
}	
}	
// Comment : Information about neighbor logon carriers (optional)	
if (opt_has_FWD_neighbor_infos_flag == '1') {	
for (l=0; l < N_neighbors_FWD; l++) {	
If ((l % 2)==0) {	
FWD_carrier_frequency	16
FWD_carrier_type	4
} else {	
FWD_carrier_type	4
FWD_carrier_frequency	16
}	
if (N_neighbors_FWD % 2 ==1) {	
reserved	4
}	
}	

} minimal Length	304
Length as encoded	800

The message content is built from the following information:

- **SigMsgID** (8 bits): to identify the signalling message and shall be equal to 0x11.
- **opt_has_FWD_neighbor_infos_flag** (1bit): Flag to identify the presence of FWD neighbors option.
- **RTN_channel_definition_per_NCC_flag** (1 bit): If multiple NCC entities are supported in the SSP network, flag indicating whether the NCC will specify its own subset of the RTN link channels indicated below. A UT wishing to logon to this specific NCC uses only this subset of channel definitions.
- **NCC_list_number** (4bits): If the system supports multiple NCCs (distributed control), a NCC may include in its logon table several sets of logon config parameters, with each set associated with a different NCC (up to NCC_list_number).
- **N_neighbors_FWD** (4 bits): Number of neighbour carriers identified. The information on the neighbour carriers is included as an optional part.
- **N_max_csp_ids** (4 bits): Maximum number of CSP identified per NCC. Equal to the number of CSP ids for single NCC. Maximum value is 15.
- **N_CC_cong_states** (8 bits): Number of congestion control traffic states. Default value is 4.
- **N_CC_parameter_sets** (4 bits): Number of congestion control parameters sets. It has a minimum value of 1 to include the configuration to send the LOGON REQUEST, but a higher value can be used.
- **N_RACH_burst_configurations** (4 bits): Number of RACH burst configurations parameters sets that are described in this message.
- **Number_preamble_scrambling_options[m]** (3 bits): Number of preamble and scrambling options defined for each RACH burst configuration (m) defined in this message.
- **N_RTN_channels** (8 bits): Number of RTN channels defined in this message.
- **N_supported_RACH_confs[n]** (4 bits): Number of supported RACH burst configurations for each RTN channel (n).
- **N_RTN_channels_per_NCC**: Number of RTN link channels that will be defined by the NCC using its own subset of channels (indicated by the *RTN_channel_definition_per_NCC* flag).
- **Beam_id** (8 bits): Beam identifier of the beam used by the FWD link carrier broadcasting the logon table. A different value shall be assigned to each beam in the mobile link per SSP network. INTEGER(0..255).

- **System_config_version_number** (8 bits): Version number of system configuration settings. This value shall be changed as a minimum whenever RACH burst configurations are modified.
- **Administrative_parameters_sys_id** (4 bits): System identifier, provided for informative purposes. A different value shall be assigned to each system by the relevant authority.
- **Administrative_parameters_ssp_id** (4 bits): SSP identifier, provided for informative purposes. A different value shall be assigned to each SSP within the same system by the relevant authority.
- **FWD_link_band_central_frequency (16 bits)**: Central frequency of the whole FWD link band used by the system (mobile link). Specified in units of 1MHz. Unsigned integer. This value is used together with the FWD_carrier_frequency parameter (defined as an offset from this frequency) to derive actual FWD link carrier frequencies.
- **RTN_link_band_central_frequency (16 bits)**: Central frequency of the whole RTN link band used by the system (mobile link). Specified in units of 1MHz. Unsigned integer. This value is used together with the RTN_channel_frequency parameter (defined as an offset from this frequency) to derive actual RTN link channel centre frequencies.
- **CC_config(N_CC_cong_states)**: Structure including the congestion control parameters to be used to access the system. One of these structures shall include the configuration parameters needed to send the LOGON REQUEST signalling message. Additional structures may be used to include a more complete set of configuration parameters. Refer to the common structure definition in section 11.10.
- **RACH_ burst_configuration()**: Structure including general RTN link RACH burst configuration parameters to be used to access the system. Each structure is called with the parameter Number_preamble_scrambling_options of the corresponding configuration. At least one structure shall be present to include the configuration parameters needed tp send the LOGON REQUEST. Additional structure may be used to include a more complete set of configurations. Refer to the common structure definition in section section 11.10.
- **RTN_channel_configuration()**: Structure including the RTN link channel configuration parameters for a specific RTN link channel. The Logon Table may only describe the signalling channel(s) to be used to send the LOGON REQUEST signalling message or a more complete set of channels. Refer to the common structure definition in section section 11.10.
- **Traffic_status** (8 bits): Refer to the CC_STATUS message (D018-COM-FUN-3720). If this field is set to b00000000, then the UT shall wait for the reception of a CC_STATUS message in order to obtain the required traffic status information.
- **Administrative_parameters_NCC_id** (4 bits): NCC manufacturer or administrator identifier, provided for informative purposes only. A different non-zero value shall be assigned to each NCC by the relevant authority. It can be set to zero, meaning that no information is provided.
- **NCC_ID** (8 bits): The L2 destination address of the NCC, as defined in D018-COM-FUN-0450. This parameter shall be included as L2 Destination address field in the

encapsulation header of the LOGON REQUEST message sent by the UT. Refer also to D018-COM-FUN-1780.

- **Logon_request_retransmission_number** (4 bits): Maximum number of LOGON REQUEST message retransmissions, as indicated in D018-COM-FUN-2482. It is encoded as an INTEGER (0..15). A value of 15 indicates that the NCC does not restrict the maximum number of LOGON REQUEST retransmissions.
- **Administrative_parameters_CSP_id** (4 bits): CSP Identifier of the CSP supported by a certain NCC, provided for informative purposes. A different non-zero value shall be assigned to each CSP by the relevant authority. In the event that the number of CSP supported by an NCC is less than N_max_csp_ids, an ID of b0000 must be used.
- **RTN_channel_id** (8 bits): Value of the *RTN_channel_id* field included in the *RTN_channel_configuration* structure of the referenced RTN link channel. Refer to the common structure definition in section 11.10.
- **FWD_carrier_frequency (16 bits)**: Central frequency of a logon carrier in a neighbour beam, provided to ease handover detection. Specified as an offset in steps of 1kHz from the FWD_link_band_central_frequency parameter defined above. Signed integer.
- **FWD_carrier_type (4 bits)**: FWD carrier type of a logon carrier in a neighbour beam. This parameter can take two values in the current version of the CS:

FWD_carrier_type	Value	Comment
Normal rate waveform	0x00	As defined in section 7.1
Low rate waveform	0x01	As defined in section 12
For future use	Any other value	

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: NCC_ID (of the NCC broadcasting the LOGON TABLE message)
- Destination address: 0xFFFF (broadcast address)

In the table below, the following notation is used to indicate the flags:

- FNf: opt_has_FWD_neighbor_infos_flag
- pNf: RTN_channel_definition_per_NCC_flag
- R: reserved
- N_neighb_FWD: N_neighbors_FWD
- #CC_tr_states: N_CC_cong_states
- #CCsets : N_CC_parameter_sets
- ps_opts#0 : Number_preamble_scrambling_options [m=0]

- supRACHchRTN#0: N_supported_RACH_confs [n=0]
- sys_id : Administrative_parameters_sys_id
- ssp_id: Administrative_parameters_ssp_id

The encoding of the message shall follow the format described below.

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x11)</td><td>ID</td><td rowspan="8">Optional & variable fields</td></tr>
<tr><td>2</td><td>FNf</td><td>pNf</td><td colspan="2">R</td><td colspan="4">NCC_list_number</td><td>Options Flags</td></tr>
<tr><td>3</td><td colspan="4">N_max_CSP_ids</td><td colspan="4">N_neighb_FWD</td><td rowspan="6">Length</td></tr>
<tr><td>4</td><td colspan="8">#CC_tr_states</td></tr>
<tr><td>5</td><td colspan="4">#CCsets</td><td colspan="4">Nr_RACH_cfs</td></tr>
<tr><td>6</td><td colspan="3">ps_opts#0</td><td colspan="3">ps_opts#1</td><td colspan="2">R</td></tr>
<tr><td>7</td><td colspan="8">Number_RTN_channels</td></tr>
<tr><td>8</td><td colspan="4">supRACHchRTN#0</td><td colspan="4">supRACHchRTN#1</td></tr>
<tr><td>9</td><td colspan="8">N_RTN_channels_per_NCC</td><td>Optional infos</td><td>Option</td></tr>
<tr><td>10</td><td colspan="8">Beam_id</td><td rowspan="2">General infos</td><td rowspan="3">General infos</td></tr>
<tr><td>11</td><td colspan="8">System_config_version_number</td></tr>
<tr><td>12</td><td colspan="4">sys_id</td><td colspan="4">ssp_id</td><td>Administrative</td></tr>
<tr><td>13</td><td colspan="8" rowspan="2">FWD_center_frequency</td><td rowspan="2">FWD Freq</td><td rowspan="4">System Frequencies</td></tr>
<tr><td>14</td></tr>
<tr><td>15</td><td colspan="8" rowspan="2">RTN_center_frequency</td><td rowspan="2">RTN Freq</td></tr>
<tr><td>16</td></tr>
<tr><td>17</td><td colspan="3">CC_cat_id</td><td colspan="2">CC_sf</td><td colspan="3">reserved</td><td>CC Type</td><td rowspan="8">RTN congestion control information</td></tr>
<tr><td>18</td><td colspan="8">tx_backoff [k=0]</td><td rowspan="3">CC Config#0</td></tr>
<tr><td>19</td><td colspan="8">persistence [k=0]</td></tr>
<tr><td>20</td><td colspan="8">retransmission_timeout [k=0]</td></tr>
<tr><td>…</td><td colspan="8">…</td><td>…</td></tr>
<tr><td>27</td><td colspan="8">tx_backoff [k=3]</td><td rowspan="3">CC Config#3</td></tr>
<tr><td>28</td><td colspan="8">persistence [k=3]</td></tr>
<tr><td>29</td><td colspan="8">retransmission_timeout [k=3]</td></tr>
<tr><td>30</td><td colspan="4">RACH conf ID</td><td colspan="4">reserved</td><td></td><td>RACH burst</td></tr>
</table>

31	DCH_channelization_code			configuration
32	ACH_channelization_code			
33	Preamble_scrambling sequence_options (N_sp_opts=8)			
…				
65				
66	RTN channel ID #0		ID	RTN channel configuration
67	RTN center frequency #0		Frequency	
68				
69	RTN Chan Type	RTN Chan Serv	Other chan info	
70	RACH ID	reserved	RACH supp	
71	Preamble_scrambling_mask			
72	Traffic_status		Status	
73	RTN channel ID #1		ID	
74	RTN center frequency #1		Frequency	
75				
76	RTN Chan Type	RTN Chan Serv	Other chan info	
77	RACH ID #0	reserved	RACH supp	
78	Preamble_scrambling_mask #0			
79	Traffic_status#0			
80	RACH ID #1	reserved		
81	Preamble_scrambling_mask #1			
82	Traffic_status#1		Status	
83	Adm_prm_NCC	Logon_retrx_num	Administrative	CNG parameters block
84	NCC_ID		Address	
85	CSP_id#0	CSP_id#1	Administrative	
86	RTN_Channel_id#0		Channel def	
87	Adm_prm_NCC	Logon_retrx_num	Administrative	CNG parameters block
88	NCC_ID		Address	
89	CSP_id#0	CSP_id#1	Administrative	
90	RTN_Channel_id#1		Channel def	
91	FWD_center_frequency [neighbor = 0]		FWD neighbors #0	Options
92				

<table>
<tr><td>93</td><td>Carr._type [n.=0]</td><td>Carr._type [n.=1]</td><td></td><td rowspan="8"></td></tr>
<tr><td>94</td><td colspan="2" rowspan="2">FWD_center_frequency
[neighbor = 1]</td><td rowspan="2">FWD neighbors
#1</td></tr>
<tr><td>95</td></tr>
<tr><td>96</td><td colspan="2" rowspan="2">FWD_center_frequency
[neighbor = 2]</td><td rowspan="2">FWD neighbors
#2</td></tr>
<tr><td>97</td></tr>
<tr><td>98</td><td>Carr._type [n.=2]</td><td>Carr._type [n.=3]</td><td rowspan="3">FWD neighbors
#3</td></tr>
<tr><td>99</td><td colspan="2" rowspan="2">FWD_center_frequency
[neighbor = 3]</td></tr>
<tr><td>100</td></tr>
</table>

11.2 Handover

The following description of Handover message will use the following general purpose syntax:

- **FWD_power_measurement_data_structure** (32 bits)**:**

Syntax	No. of bits
FWD_power_measurement_data_structure() {	
FWD_carrier_frequency	16
Primary_link_quality	8
Additional_link_quality	8
}	**32**

The FWD_power_measurement_data_structure content is built using the following information:

- **FWD_carrier_frequency** (16 bits): Same as in logon table (D018-COM-FUN-3520).
- **Primary link quality information** (8 bits): Indicator of the carrier link quality level, provided as a number in the range (0..255). The higher the number, the higher the quality.
- **Additional link quality information** (8 bits): Additional indicator of the carrier link quality level, provided as a number in the range (0..255), that can be used as secondary criteria when the primary link quality field is the same for some carriers. The higher the number, the higher the quality.

- **Position_data_structure** (96 bits):

Syntax	No. of bits
position_data_structure() {	
Latitude	32
Longitude	32
Altitude	16

Course	16
}	**96**

The position_data_structure content is built using the following information, referred to as WGS 84:

- **Latitude** (32 bits) Latitude coordinate in degrees. Float.
- **Longitude** (32bits) Longitude coordinate in degrees. Float.
- **Altitude** (16 bits): Altitude in meters. Signed integer.
- **Course** (16 bits): Course North reference clockwise in degrees. Unsigned integer.

Note: The FLOAT value shall follow the IEEE 754 standard (IEEE Standard for Binary Floating-Point Arithmetic (ANSI/IEEE Std 754-1985)). Concerning the byte order, please conform to integer byte order requirements specified in D018-COM-FUN-0410 and D018-COM-FUN-0420; big endian shall be used.

11.2.1 HO Recommendation

D018-COM-FUN-3530

The HO Recommendation message shall comply with the following structure:

Syntax	No. of bits
HORecomm_payload_structure() {	
SigMsgID (0x22)	8
Number_positions	8
HO_recommendation_type	8
HO_request_id_UT	4
HO_request_id_GS	4
for (i=0; i < 4; i++) {	
FWDpowermeasurement_data_structure (neighbor=i)	32
}	
if (Number_positions > 0) {	
for (j=0; j < Number_positions; j++) {	
position_data_structure (trajectory = j)	96
}	
}	

}	without Options	**160**
	with All Options	**448**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message. Equal to 0x22 for HO Recommendation message.
- **Number_positions**(8 bits): Number of trajectory points reported in the message. If zero, no trajectory points are included. Maximum value is 3.
- **HO_recommendation_type** (8 bits): Type of handover recommendation. The following types are defined: signal-based, GS-requested, manual, APB, position-based. The following encoding must be used:

HO_recommendation_type	Value
Signal-based	0x01
GS-requested	0x02
Manual	0x03
APB	0x04
Position-based	0x05
For future use	Any other value

- **Ho_request_id_GS** (4 bits): HO operation identifier (counter), for GS-triggered HO operations.
    - For UT-triggered HOs, it is set to 0.
    - For GS-triggered HOs, the GS sets this value in an initial HO INFO REQUEST or HO COMMAND message and the UT uses this value for all HO signalling messages associated with the same HO operation.
- **Ho_request_id_UT** (4 bits): HO operation identifier (counter), for UT-triggered HO operations.
    - For GS-triggered HOs, it is set to 0.
    - For UT-triggered HOs, the UT sets this value in an initial HO RECOMMENDATION message and the GS uses this value for all HO signalling messages associated with the same HO operation.

  These last two fields together provide an identifier for a given HO operation. Refer to HO_REQ_ID field in D018-COM-FUN-2761, D018-COM-FUN-2781 and D018-COM-FUN-2791 above.
- **FWDpowermeasurement_data_structure** (32 bits): as described previously.

- **position_data_structure** (96 bits): as described previously.

In the table below, the following notation is used to indicate the flags:

- HORecomm_Type: HO_recommendation_type
- HO_req_id_UT: HO_request_id_UT
- HO_req_id_GS: HO_request_id_GS

The encoding of the message follows the format described below.

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x22)</td><td>ID</td><td rowspan="2">Optional & variable fields</td></tr>
<tr><td>2</td><td colspan="8">Number_positions</td><td>flags/length</td></tr>
<tr><td>3</td><td colspan="8">HORecomm_Type</td><td>HO Type</td><td rowspan="18">General infos</td></tr>
<tr><td>4</td><td colspan="4">HO_req_id_UT</td><td colspan="4">HO_req_id_GS</td><td>Rqst Id</td></tr>
<tr><td>5</td><td colspan="8" rowspan="4">Measurement (Neighbor = 1)</td><td rowspan="16">FWD Power Measurement</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td></tr>
<tr><td>8</td></tr>
<tr><td>9</td><td colspan="8" rowspan="4">Measurement (Neighbor = 2)</td></tr>
<tr><td>10</td></tr>
<tr><td>11</td></tr>
<tr><td>12</td></tr>
<tr><td>13</td><td colspan="8" rowspan="4">Measurement (Neighbor = 3)</td></tr>
<tr><td>14</td></tr>
<tr><td>15</td></tr>
<tr><td>16</td></tr>
<tr><td>17</td><td colspan="8" rowspan="4">Measurement (Neighbor =4)</td></tr>
<tr><td>18</td></tr>
<tr><td>19</td></tr>
<tr><td>20</td></tr>
<tr><td>21</td><td colspan="8" rowspan="12">Trajectory Point#1
Latitude
Longitude
Altitude
Course</td><td rowspan="12">Trajectory Point 1 (position)</td><td rowspan="12">Trajectory (Optional)</td></tr>
<tr><td>22</td></tr>
<tr><td>23</td></tr>
<tr><td>24</td></tr>
<tr><td>25</td></tr>
<tr><td>26</td></tr>
<tr><td>27</td></tr>
<tr><td>28</td></tr>
<tr><td>29</td></tr>
<tr><td>30</td></tr>
<tr><td>31</td></tr>
<tr><td>32</td></tr>
</table>

33	…	…	
…			
44			
45	Trajectory Point #3	Trajectory Point 3	
…			
56			

The encapsulation header also includes the following field(s) with the specified value:

- Source address: UT_ID
- Destination address: NCC_ID

11.2.2 HO Command

D018-COM-FUN-3540

The HO Command message shall comply with the following structure:

Syntax	No. of bits
HOCmd_payload_structure() {	
SigMsgID (0x23)	8
opt_has_no_RACH_burst_config_flag	1
opt_has_control_code_flag	1
opt_HO_finished_message_required_flag	1
opt_has_FWD_neighbor_infos_flag	1
// comment: new RACH burst configuration	
N_RACH_burst_configurations	4
Number_RTN_channels	8
for (n=0; n < N_RTN_channels; n++) {	
//Comment: for each RTN Channel	
N_supported_RACH_confs[n]	4
}	
if (Number_RTN_channels % 2 ==1) {	
reserved	4
}	
N_CC_cong_states	8

N_CC_parameter_sets	4
for (m=0; m < N_RACH_burst_configurations; m++) {	
// comment: for each additional RACH burst conf	
Number_preamble_scrambling_options[m]	3
}	
if (((3*N_RACH_burst_configurations+4) % 8) !=0) {	
reserved (length= 8-(3*N_RACH_burst_configurations+4)%8)	8
}	
N_neighbors_FWD	4
reserved	4
HO_type	8
HO_request_id_UT	4
HO_request_id_GS	4
HO_result_valid	8
UT_ID_new	16
GES_ID_new	8
NCC_ID_new	8
// Comment: Physical_layer_parameters	
for (l=0; l < Number_RTN_carriers; l++) {	
RTN_channel_configuration (N_supported_RACH_confs[l])	
Traffic_status	8
}	
// Comment: RTN Congestion Information	
// Comment: New CC config	
for (i=0 ; i < N_CC_parameter_sets; i++) {	
CC_config(N_CC_cong_states)	
}	
FWD_carrier_frequency	16
FWD_carrier_type	4
// Comment: other configuration parameters	
generic_retransmission_number	4
old_channel_empty_timeout	8

L3_update_timeout	8
Connection_close_timeout	8
FLC_close_timeout	8
No_received_data_timeout	8
if (opt_has_no_RACH_burst_config_flag == '1') {	
// Comment: new RACH burst configurations	
for (m=0; m < N_RACH_burst_configurations; m++) {	
RACH burst_configuration (Number_preamble_scrambling_options[m])	
}	
}	
// Comment: control code (optional)	
if (opt_has_control_code_flag == '1') {	
control_code_FWD_carrier_id	128
}	
// Comment : Information of neighbor logon carriers (optional)	
if (opt_has_FWD_neighbor_infos_flag == '1') {	
for (l=0; l < N_neighbors_FWD; l++) {	
FWD_carrier_frequency	16
FWD_carrier_type	4
}	
}	
}	
// Comment : Information of neighbor logon carriers (optional)	
if (opt_has_FWD_neighbor_infos_flag == '1') {	
for (l=0; l < N_neighbors_FWD; l++) {	
If ((l % 2)==0) {	
FWD_carrier_frequency	16
FWD_carrier_type	4
} else {	
FWD_carrier_type	4
FWD_carrier_frequency	16
}	

if (N_neighbors_FWD % 2 ==1) {	
reserved	4
}	
}	
} Length as encoded	1032

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message. Equal to 0x23 for HO Command message.
- **opt_has_FWD_neighbor_infos_flag** (1bit): Flag to identify the presence of FWD neighbors option.
- **opt_has_no_RACH_burst_config_flag** (1 bit): Flag to indicate the presence of RACH burst configurations.
- **opt_has_control_code_flag** (1 bit): Flag to indicate that a control code is present in the message.
- **opt_HO_finished_message_required_flag** (1 bit): If set to one, the HO FINISHED message shall be sent by the UT to the NCC as the final step in the HO procedure. This is the flag referred to as "*GS WAN*" in D018-COM-FUN-2761 and D018-COM-FUN-2781 above.
- **HO_type** (8 bits)**:** type of handover. The following encoding has to be used:

HO_type	**Value**
SSP	0x01
GES	0x02
Fast HO	0x03
For future use	Any other value

- **HO_request_id_GS** (4 bits): See HO Recommendation message D018-COM-FUN-3530.
- **HO_request_id_UT** (4 bits): See HO Recommendation message D018-COM-FUN-3530.
- **HO_Result _valid** (8 bits)**:** HO recommendation result. 0 for accepted, 1 for rejected.
- **UT_ID_new** (16 bits): UT L2 Address to be used over the new link. Refer to Logon initial accept message D018-COM-FUN-3650.
- **GES_ID_new** (8 bits): New GES L2 Address of the new GES. Refer to Logon initial accept message D018-COM-FUN-3650.
- **NCC_ID_new** (8 bits): NCC L2 Address of the new NCC. Refer to Logon Table message D018-COM-FUN-3520.

Refer to D018-COM-FUN-3650 (LOGON INITIAL ACCEPT) and D018-COM-FUN-3520 above (LOGON TABLE) for the description of the remaining fields.

The encoding of the message follows the format described below.

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x23)</td><td>ID</td><td rowspan="8">Optional & variable fields</td></tr>
<tr><td>2</td><td>Rf</td><td>Cf</td><td>Ff</td><td>Nf</td><td colspan="4">Nr_RACH_cfs</td><td>Options Flags</td></tr>
<tr><td>3</td><td colspan="8">Number_RTN_channels</td><td rowspan="6">Length</td></tr>
<tr><td>4</td><td colspan="4">supRACHchRTN#0</td><td colspan="4">supRACHchRTN#1</td></tr>
<tr><td>5</td><td colspan="8">#CCsets</td></tr>
<tr><td>6</td><td colspan="4">#CCsets</td><td colspan="3">ps_opts#0</td><td>ps_</td></tr>
<tr><td>7</td><td colspan="2">opts#1</td><td colspan="3">ps_opts#2</td><td colspan="3">ps_opts#3</td></tr>
<tr><td>8</td><td colspan="4">N_neighb_FWD</td><td colspan="4">reserved</td></tr>
<tr><td>9</td><td colspan="8">HO_type</td><td>HO Type</td><td rowspan="13">General infos</td></tr>
<tr><td>10</td><td colspan="4">HO_request_id_UT</td><td colspan="4">HO_request_id_GS</td><td>HO id</td></tr>
<tr><td>11</td><td colspan="8">HO_result_valid</td><td>HO result</td></tr>
<tr><td>12</td><td colspan="8" rowspan="6">GES_Satellite_MAC_address</td><td rowspan="10">Address</td></tr>
<tr><td>13</td></tr>
<tr><td>14</td></tr>
<tr><td>15</td></tr>
<tr><td>16</td></tr>
<tr><td>17</td></tr>
<tr><td>18</td><td colspan="8" rowspan="2">UT_ID_new</td></tr>
<tr><td>19</td></tr>
<tr><td>20</td><td colspan="8">GES_ID_new</td></tr>
<tr><td>21</td><td colspan="8">NCC_ID_new</td></tr>
<tr><td>22</td><td colspan="8">RTN channel ID #0</td><td>ID</td><td rowspan="6">RTN Channel Config</td></tr>
<tr><td>23</td><td colspan="8" rowspan="2">RTN center frequency #0</td><td rowspan="2">Frequency</td></tr>
<tr><td>24</td></tr>
<tr><td>25</td><td colspan="4">RTN Chan Type</td><td colspan="4">RTN Chan Serv</td><td>Other ch. info</td></tr>
<tr><td>26</td><td colspan="4">RACH ID</td><td colspan="4">Traffic_status</td><td rowspan="2">RACH supp</td></tr>
<tr><td>27</td><td colspan="8">Preamble_scrambling_mask</td></tr>
<tr><td>28</td><td colspan="8">RTN channel ID #1</td><td>ID</td><td rowspan="8">RTN Channel Config</td></tr>
<tr><td>29</td><td colspan="8" rowspan="2">RTN center frequency #1</td><td rowspan="2">Frequency</td></tr>
<tr><td>30</td></tr>
<tr><td>31</td><td colspan="4">RTN Chan Type</td><td colspan="4">RTN Chan Serv</td><td>Other ch. info</td></tr>
<tr><td>32</td><td colspan="4">RACH ID #0</td><td colspan="4">Traffic_status#0</td><td rowspan="4">RACH supp</td></tr>
<tr><td>33</td><td colspan="8">Preamble_scrambling_mask #0</td></tr>
<tr><td>34</td><td colspan="4">RACH ID #1</td><td colspan="4">Traffic_status#1</td></tr>
<tr><td>35</td><td colspan="8">Preamble_scrambling_mask #1</td></tr>
<tr><td>36</td><td colspan="3">CC_cat_id</td><td colspan="2">CC_sf</td><td colspan="3">reserved</td><td>CC Type</td><td rowspan="6">RTN congestion control information</td></tr>
<tr><td>37</td><td colspan="8">tx_backoff [k=0]</td><td rowspan="3">CC Config#0</td></tr>
<tr><td>38</td><td colspan="8">persistence [k=0]</td></tr>
<tr><td>39</td><td colspan="8">retransmission_timeout [k=0]</td></tr>
<tr><td>…</td><td colspan="8">…</td><td>…</td></tr>
<tr><td>46</td><td colspan="8">tx_backoff [k=3]</td><td>CC Config#3</td></tr>
</table>

<table>
<tr><td>47</td><td colspan="3">persistence [k=3]</td><td rowspan="2"></td><td rowspan="2"></td></tr>
<tr><td>48</td><td colspan="3">retransmission_timeout [k=3]</td></tr>
<tr><td>49</td><td colspan="3" rowspan="2">FWD_carrier_frequency</td><td rowspan="3">FWD infos</td><td rowspan="3">FWD Carrier Info</td></tr>
<tr><td>50</td></tr>
<tr><td>51</td><td>FWD_carr._type</td><td colspan="2">rtx_Number</td></tr>
<tr><td>52</td><td colspan="3">old_channel_empty_timeout</td><td rowspan="5">Configuration parameters</td><td rowspan="5">Configuration parameters</td></tr>
<tr><td>53</td><td colspan="3">L3_update_timeout</td></tr>
<tr><td>54</td><td colspan="3">Connection_close_timeout</td></tr>
<tr><td>55</td><td colspan="3">FLC_close_timeout</td></tr>
<tr><td>56</td><td colspan="3">No_received_data_timeout</td></tr>
<tr><td>57</td><td>RACH conf ID</td><td>R</td><td>N_sp_opts</td><td rowspan="6">RACH burst conf #0</td><td rowspan="14">RACH info (Optional)</td></tr>
<tr><td>58</td><td colspan="3">DCH_channelization_code</td></tr>
<tr><td>59</td><td colspan="3">ACH_channelization_code</td></tr>
<tr><td>60</td><td colspan="3" rowspan="3">Preamble_scrambling sequence_options</td></tr>
<tr><td>…</td></tr>
<tr><td>92</td></tr>
<tr><td>93</td><td>RACH conf ID</td><td>R</td><td>N_sp_opts</td><td rowspan="8">RACH burst conf #1</td></tr>
<tr><td>94</td><td colspan="3">DCH_channelization_code</td></tr>
<tr><td>95</td><td colspan="3">ACH_channelization_code</td></tr>
<tr><td>96</td><td colspan="3" rowspan="2">Preamble Sequence</td></tr>
<tr><td>97</td></tr>
<tr><td>98</td><td colspan="3" rowspan="3">Scrambling Sequence</td></tr>
<tr><td>99</td></tr>
<tr><td>100</td></tr>
<tr><td>101</td><td colspan="3" rowspan="3">control_code_with_FWD_carrier_id</td><td rowspan="3">Control Code</td><td rowspan="3">Control Code</td></tr>
<tr><td>…</td></tr>
<tr><td>116</td></tr>
<tr><td>117</td><td colspan="3" rowspan="2">FWD_center_frequency [neighbor = 0]</td><td rowspan="3">FWD neighbors #0</td><td rowspan="10">Options</td></tr>
<tr><td>118</td></tr>
<tr><td>119</td><td>Carr._type [n.=0]</td><td colspan="2">Carr._type [n.=1]</td></tr>
<tr><td>120</td><td colspan="3" rowspan="2">FWD_center_frequency [neighbor = 1]</td><td rowspan="2">FWD neighbors #1</td></tr>
<tr><td>121</td></tr>
<tr><td>122</td><td colspan="3" rowspan="2">FWD_center_frequency [neighbor = 2]</td><td rowspan="2">FWD neighbors #2</td></tr>
<tr><td>123</td></tr>
<tr><td>124</td><td>Carr._type [n.=2]</td><td colspan="2">Carr._type [n.=3]</td><td rowspan="3">FWD neighbors #3</td></tr>
<tr><td>125</td><td colspan="3" rowspan="2">FWD_center_frequency [neighbor = 3]</td></tr>
<tr><td>126</td></tr>
</table>

In the previous table, the following notation is used to indicate the flags:

- Rf : opt_has_no_predefined_RTN_carriers_flag
- Cf : opt_has_control_code_flag
- Ff: opt_has_FWD_neighbor_infos_flag
- Nf: opt_HO_finished_message_required_flag
- Nr_RACH_cfs: N_RACH_burst_configurations
- supRACHchRTN: N_supported_RACH_confs

- #CC_tr_states: N_CC_cong_states
- #CCsets : N_CC_parameter_sets

ps_opts#0: Number_preamble_scrambling_options[m=0]
The encapsulation header also includes the following field(s) with the specified value:

- Source address: NCC_ID
- Destination address: UT_ID

11.2.3 ACK HO Command

D018-COM-FUN-3550

The ACK HO Command shall comply with the following structure:

Syntax	No. of bits
HOACKCmd_payload_structure() {	
SigMsgID (0x24)	8
HO_request_id_UT	4
HO_request_id_GS	4
HO_result	8
}	**24**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message. Equal to 0x24 for ACK HO Command message.
- **HO_request_id_GS** (4 bit): See D018-COM-FUN-3530.
- **HO_request_id_UT** (4 bit): See D018-COM-FUN-3530.
- **HO_result** (8 bit): HO result. Result. 0x00: HO accepted. All other values: HO rejected.

The encoding of the message follows the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x24)</td><td>ID</td><td>Opt. & var. Flds</td></tr>
<tr><td>2</td><td colspan="4">HO_request_id_UT</td><td colspan="4">HO_request_id_GS</td><td>Rqst Id</td><td rowspan="2">General infos</td></tr>
<tr><td>3</td><td colspan="8">HO_result</td><td>Result</td></tr>
</table>

The encapsulation header also includes the following field(s) with the specified value:

- Source address: UT_ID
- Destination address: NCC_ID

11.2.4 Connection close

D018-COM-FUN-3560

The Connection close message shall comply with the following structure:

Syntax	No. of bits
HOConnClose_payload_structure() {	
MsgID (0x29)	8
HO_conn_close_result	8
}	**16**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message. Equal to 0x29 for Connection close message.
- **HO_conn_close_result** (8 bits):

HO_conn_close_result	Value
Nominal	0x00
Timeout exceeded	0x01
For future use	Any other value

The encoding of the message follows the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x29)								ID	Opt. & var. Flds
2	HO_conn_close_result								Result	General infos

The encapsulation header also includes the following field(s) with the specified value:

- Source address: UT_ID
- Destination address: GES_ID

11.2.5 ACK Connection close

D018-COM-FUN-3570

The ACK Connection close message shall comply with the following structure:

Syntax	No. of bits
HOConnCloseACK_payload_structure() {	
MsgID (0x2A)	8
}	**8**

The message content is built from the following information:

- **SigMsgID** (8 bits): to identify the signalling message. Equal to 0x2A for ACK Connection close message.

The encoding of the message follows the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x2A)								ID	Opt. & var. Flds

The encapsulation header also includes the following field(s) with the specified value:

- Source address: GES_ID
- Destination address: UT_ID

11.2.6 HO Finished

D018-COM-FUN-3580

The HO Finished message shall comply with the following structure:

Syntax	No. of bits
HOFinished_payload_structure() {	
MsgID (0x2B)	8
HO_request_id_UT	4
HO_request_id_GS	4

HO_result_failure_code	8
}	**24**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message. Equal to 0x2B for HO Finished message.
- **HO_request_id_GS** (4 bit): See D018-COM-FUN-3530.
- **HO_request_id_UT** (4 bit): See D018-COM-FUN-3530.
- **HO_result_failure_code** (8 bits):

HO_result_failure_code		**Value**
Result	Reason	
HO performed	HO successful	0x00
	Timeout emptying buffer	0x01 (1d)
	Timeout closing channel	0x02 (2d)
	Old channel lost	0x03 (3d)
	Any other value (future use)	< 0x80
HO failed not performed	Logon Rejected	0x80 (128d)
	Logon no answer	0x81 (129d)
	Validation rejected	0x82 (130d)
	No answer from GES	0x83 (131d)
	New channel lost	0x84 (132d)
	Any other value (future use)	>= 0x85

The encoding of the message follows the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x2B)</td><td>ID</td><td>Opt. & var. Flds</td></tr>
<tr><td>2</td><td colspan="4">HO_request_id_UT</td><td colspan="4">HO_request_id_GS</td><td rowspan="2">General infos</td><td rowspan="2">General infos</td></tr>
<tr><td>3</td><td colspan="8">HO_result_failure_code</td></tr>
</table>

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: UT_ID
- Destination address: NCC_ID

11.2.7 ACK HO Finished

D018-COM-FUN-3590

The ACK HO Finished message shall comply with the following structure:

Syntax	No. of bits
HOFinishedACK_payload_structure() {	
MsgID (0x2C)	8
HO_request_id_UT	4
HO_request_id_GS	4
}	**16**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message. Equal to 0x2C for ACK HO Finished message.
- **HO_request_id_GS** (4 bit): See D018-COM-FUN-3530.
- **HO_request_id_UT** (4 bit): See D018-COM-FUN-3530.

The encoding of the message follows the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x2C)</td><td>ID</td><td>Opt. & var. Flds</td></tr>
<tr><td>2</td><td colspan="4">HO_request_id_UT</td><td colspan="4">HO_request_id_GS</td><td>Gal infos</td><td>Gal infos</td></tr>
</table>

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: NCC_ID
- Destination address: UT_ID

11.2.8 HO Request Info

D018-COM-FUN-3591

The HO Request Info message shall comply with the following structure:

Syntax	No. of bits
HORqstInfo_payload_structure() {	
SigMsgID (0x21)	8
Ho_request_id_GS	4
Ho_request_id_UT	4
}	**16**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message. Equal to 0x21 for HO Request Info message.
- **HO_request_id_GS** (4 bit): See D018-COM-FUN-3530.
- **HO_request_id_UT** (4 bit): See D018-COM-FUN-3530.

The encoding of the message follows the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x21)								ID	Opt. & var. Flds
2	Ho_request_id_UT				Ho_request_id_GS				General infos	General infos

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: NCC_ID
- Destination address: UT_ID

11.2.9 HO Validation Request

D018-COM-FUN-3592

The HO Validation Request message shall comply with the following structure:

Syntax	No. of bits
HOvalidRQST_payload_structure() {	

SigMsgID (0x27)	8
opt_has_CSP_ids_flag	1
opt_has_control_code_flag	1
Number_Satellite_MAC_address	2
N_csp_ids	4
HO_type	8
HO_request_id_UT	4
HO_request_id_GS	4
HO_result_valid	8
Administrative_parameters_UT_manufacturer_id	4
ROHC_supported_profile	8
preferred_MODCOD	4
NCC_ID	8
UT_ICAO_ID	24
for (i=0; i < Number_Satellite_MAC_address; i++) {	
UT_Satellite_MAC_address [i]	48
}	
if (opt_has_CSP_ids_flags)	
for (j=0; j < N_csp_ids; j++) {	
preferred_CSP_id	4
}	
if (N_csp_ids % 2== 1) {	
reserved	4
}	
}	
// Comment : Physical_layer_parameters	
if (opt_has_control_code_flag == '1') {	
control_code_with_FWD_carrier_id	128
}	
} without Options	**126**
with All Options	**274**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message. Equal to 0x27 for HO Validation Request message.
- **HO_type** (8 bits): See D018-COM-FUN-3540**.**
- **HO_request_id_GS** (4 bit): See D018-COM-FUN-3530.
- **HO_request_id_UT** (4 bit): See D018-COM-FUN-3530.
- **HO_result_valid** (8 bit): Result. 0x00: HO accepted. All other values: HO rejected.

Refer to D018-COM-FUN-3660 (LOGON VALIDATION REQUEST) for the description of the remaining fields.

The encoding of the message follows the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x27)</td><td>ID</td><td>Optional & variable</td></tr>
<tr><td>2</td><td>Cf</td><td>CPf</td><td colspan="2">N_addr</td><td colspan="4">N_csp_ids</td><td>Options Flags</td><td></td></tr>
<tr><td>3</td><td colspan="8">HO_type</td><td>Rqst Id</td><td rowspan="15">General infos</td></tr>
<tr><td>4</td><td colspan="4">HO_request_id_UT</td><td colspan="4">HO_request_id_GS</td><td>HO Type</td></tr>
<tr><td>5</td><td colspan="8">HO_result_valid</td><td>Result</td></tr>
<tr><td>6</td><td colspan="4">Adm_UT_ID</td><td colspan="4">prefMODCOD</td><td rowspan="2">Gal Infos</td></tr>
<tr><td>7</td><td colspan="8">ROHC_supported_profile</td></tr>
<tr><td>8</td><td colspan="8">NCC_ID</td><td rowspan="4">Additional Addresses</td></tr>
<tr><td>9</td><td colspan="8" rowspan="3">UT_ICAO_ID</td></tr>
<tr><td>10</td></tr>
<tr><td>11</td></tr>
<tr><td>12</td><td colspan="8" rowspan="6">UT_Satellite_
MAC_address</td><td rowspan="6">Address</td></tr>
<tr><td>13</td></tr>
<tr><td>14</td></tr>
<tr><td>15</td></tr>
<tr><td>16</td></tr>
<tr><td>17</td></tr>
<tr><td>18</td><td colspan="4">pr_CSP_id#0</td><td colspan="4">pr_CSP_id#1</td><td>Administrative</td><td rowspan="4">Options</td></tr>
<tr><td>19</td><td colspan="8" rowspan="3">control_code_with_FWD_carrier_id</td><td rowspan="3">Control Code</td></tr>
<tr><td>…</td></tr>
<tr><td>34</td></tr>
</table>

In the previous table, the following notation is used to indicate the flags:

- Cf: opt_has_control_code_flag
- CPf: opt_has_CSP_ids_flag
- N_addr: Number_UT_MAC_address_for_IP
- R: reserved

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: UT_ID
- Destination addresss: GES_ID

11.2.10 HO Validation

D018-COM-FUN-3593

The HO Validation message shall comply with the following structure:

Syntax	No. of bits
HOvalid_payload_structure() {	
SigMsgID (0x28)	8
Number_OSI_address	4
ROHC_used_flag	1
opt_has_control_code_flag	1
include_destination_flag	1
reserved	1
HO_request_id_UT	4
HO_request_id_GS	4
HO_result_valid	8
ARQ_support_per_flow_mask	16
rl_path_and_processing_time	8
GES_Satellite_MAC_address	48
// Comment: OSI_join_parameters	
for (i=0; i < Number_OSI_address; i++) {	
8208-AGR-Address [i]	48
CSP_associated [i]	4
}	
if (Number_OSI_address % 2 == 1) {	
reserved	4
}	
// GES Configuration	
ROHC_supported_profile	8

if (opt_has_control_code_flag == '1') {	
control_code_for_NCC	128
}	
} without Options	**168**
} with all Options	**296**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message. Equal to 0x28 for HO Validation message.
- **HO_type** (8 bits): See D018-COM-FUN-3540.
- **HO_request_id_GS** (4 bit): See D018-COM-FUN-3530.
- **HO_request_id_UT** (4 bit): See D018-COM-FUN-3530.
- **HO_result_valid** (8 bit): Result. 0x00: HO accepted. All other values: HO rejected.

Refer to D018-COM-FUN-3670 (LOGON VALIDATION ACCEPT) for the description of the remaining fields.

In the table below, the following notation is used to indicate the flags:

- Rf : ROHC_used _flag
- Cf: opt_has_control_code_flag
- Df: include_destination_flag
- R: reserved
- N_OSI_addr: Number_UT_MAC_address_for_IP
- CSP_ass : CSP_associated

The encoding of the message follows the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x28)</td><td>ID</td><td rowspan="2">Optional & variable fields</td></tr>
<tr><td>2</td><td>Rf</td><td>Cf</td><td>Df</td><td>R</td><td colspan="4">N_OSI_addr</td><td>flags/length</td></tr>
<tr><td>3</td><td colspan="4">HO_request_id_UT</td><td colspan="4">HO_request_id_GS</td><td rowspan="2">HO infos</td><td rowspan="9">General infos</td></tr>
<tr><td>4</td><td colspan="8">HO_result_valid</td></tr>
<tr><td>5</td><td colspan="8" rowspan="2">ARQ_support_per_flow_mask</td><td rowspan="3">General infos</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td><td colspan="8">rl_path_and_processing_time</td></tr>
<tr><td>8</td><td colspan="8" rowspan="4">GES_Satellite_MAC_address</td><td rowspan="4">OSI Addresses</td></tr>
<tr><td>9</td></tr>
<tr><td>10</td></tr>
<tr><td>11</td></tr>
</table>

12				
13				
14	8208-AGR-Address			
…				
19				
20	CSP_ass #0	reserved		
21	ROHC_supported_profile		Head. Comp	
22	control_code _for_NCC		Control Code	Options
…				
37				

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: GES_ID
- Destination address: UT_ID

11.3 Multicast

Requirements removed.

11.4 Terminal registration (logon/logoff)

*11.4.1 **Logon Request***

D018-COM-FUN-3640

The Logon Request message shall comply with the following structure:

Syntax	**No. of bits**
LOGONrequest_payload_structure() {	
SigMsgID (0x12)	8
opt_has_SSP_id_flag	1
opt_has_CSP_ids_flag	1
N_csp_ids	4
IP_support_flag	1
OSI_support_flag	1
Number_Satellite_MAC_address	2
reserved	2
Administrative_parameters_UT_id	4

Beam_id	8
UT_known_RACH_configuration_mask	8
ROHC_supported_profile	8
for (i=0; i < Number_Satellite_MAC_address; i++) {	
UT_Satellite_MAC_address [i]	48
}	
// Comment : Optional part	
if (opt_has_SSP_id_flag == '1') {	
reserved	4
prefered_SSP_id	4
}	
if (opt_has_CSP_ids_flags == '1') {	
for (j=0; j < N_csp_ids; j++) {	
preferred_CSP_id	4
}	
if (N_csp_ids % 2 ==1) {	
reserved	4
}	
}	
} without Options	**96**
with All Options	**112**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x12.
- **opt_has_SSP_id_flag** (1 bit): Flag to identify the presence of SSP Id parameter in the optional fields.
- **opt_has_CSP_ids_flag** (1 bit): Flag to identify the presence of CCP Id Parameter list in the optional fields.
- **N_csp_ids** (4 bits): Number of elements in the field preferred_CSP_id, in the order of CSP preference (i.e., the most preferred CSP first).
- **IP_support_flag** (1 bit)**:** Flag to indicate whether the IPv6 protocol stack is supported by the UT.

IP_support_flag	Value
IP not supported	0
IP supported	1

- **OSI_support_flag** (1 bit): Flag to indicate whether the OSI protocol stack is supported by the UT.

OSI_support_flag	Value
OSI not supported	0
OSI supported	1

- **Number_Satellite_MAC_address_for_IP** (2 bits): Number of addresses in the field *UT_Satellite_MAC_address*.
- **Administrative_parameters_UT_id** (4 bits): UT manufacturer identifier, provided for informative purposes only. A different non-zero value shall be assigned by the relevant authority. It can be set to zero, meaning that no information is provided.
- **Beam_id** (8 bits): Beam identifier. This field must be set by the UT to the same value as the equivalent field included in the received LOGON TABLE message.
- **UT_known_RACH_configuration_mask** (8 bits): This field is used to inform the NCC that certain RACH burst configurations, as provided through the *RACH_burst_configurations* structure, are already known by the UT for the specific system configuration version advertised in the LOGON TABLE message (*system_config_version_number* parameter). It has the form of a mask, i.e., "known config" is signalled by setting a specific bit to one, as indicated in the table below. If the UT has no configuration information or the version is no longer valid, then this field is set to zero.

Known RACH config	Associated bit in the mask
No valid information	b00000000
RACH_CR$_{160}$_SF$_{16}$_DB$_{512}$	b00000001
RACH_CR$_{160}$_SF$_{4}$_DB$_{2048}$	b00000010
RACH_CR$_{160}$_SF$_{16}$_DB$_{288}$	b00000100
RACH_CR$_{160}$_SF$_{4}$_DB$_{976}$	b00001000
Reserved	Other bits

- **ROHC_supported_profile** (8 bits): Supported ROHC profiles according to D018-COM-FUN-0620 and D018-COM-FUN-0625. This field is a mask, i.e., a supported ROHC profile is signalled by setting a specific bit to one, as indicated in the table below.

Supported ROHC	Associated bit in the

profile	mask
Reserved	b00000000
Uncompressed	b00000001
RTP	b00000010
UDP	b00000100
IP-only	b00001000
TCP	b00010000
Reserved	Other bits

- **UT_Satellite_MAC_address** (48 bits): Satellite MAC address of the UT, as defined in D018-COM-FUN-0463. Several MAC addresses can be configured by an UT (up to a maximum of 4).
- **preferred_SSP_id** (4 bits): Identifier of the SSP preferred by the UT, to be taken into account for HO decisions. It is coded in the same way as the *Administrative_parameters_SSP_id* field included in the LOGON TABLE message. Optional field.
- **preferred_CSP_id** (4 bits): Identifier of the CSP preferred by the UT, to be taken into account for GES selection. It is coded in the same way as the *Administrative_parameters_CSP_id* field included in the LOGON TABLE message. Optional field.

The encoding of the message shall follow the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x12)</td><td>ID</td><td rowspan="3">Optional & variable fields</td></tr>
<tr><td>2</td><td>SSf</td><td>CPf</td><td colspan="4">N_csp_ids</td><td>If</td><td>Of</td><td rowspan="2">Options Flags & Lengths</td></tr>
<tr><td>3</td><td colspan="2">N_MAC</td><td colspan="2">R</td><td colspan="4">Adm_UT_ID</td></tr>
<tr><td>4</td><td colspan="8">Beam_id</td><td rowspan="6">General infos</td><td rowspan="6">General infos</td></tr>
<tr><td>5</td><td colspan="8">UT_known_RACH_conf_mask</td></tr>
<tr><td>6</td><td colspan="8">ROHC_supp._profile</td></tr>
<tr><td>7</td><td colspan="8" rowspan="3">UT_Satellite_ MAC_address</td></tr>
<tr><td>8</td></tr>
<tr><td>9</td></tr>
</table>

10				
11				
12				
13	Reserved	ssp_id	Administrative	Options
14	pref_CSP_id#0	pref_CSP_id#1		

In the table above, the following notation is used to indicate the flags:

- SSf : opt_has_SSP_id_flag
- CPf : opt_has_CSP_ids_flag
- R: reserved
- If : IP_support_flag
- Of: OSI_support_flag
- N_MAC: Number_Satellite_MAC_address
- Adm_UT_ID: Administrative_parameters_UT_id

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: UT ICAO ID (24 bits address of the UT)
- Destination address: NCC_ID (of the NCC selected for logon)

11.4.2 Logon Initial Accept

D018-COM-FUN-3650

The Logon Initial Accept message shall comply with the following structure:

Syntax	No. of bits
LOGONacceptInit_payload_structure() {	
SigMsgID (0x13)	8
opt_has_no_RACH_burst_config_flag	1
opt_has_control_code_flag	1
reserved	6
Number_RTN_channels	8

for (m=0; m < Number_RTN_channels; m++) {	
//Comment: for each RTN Channel	
N_supported_RACH_confs[m]	4
}	
if (Number_RTN_channels % 2 ==1) {	
reserved	4
}	
//comment: remaining CC values	
N_CC_cong_states	8
N_CC_parameter_sets	4
// comment: additional RACH burst configuration	
N_RACH_burst_configurations	4
for (n=0; n < N_RACH_burst_configurations; n++) {	
// comment: for each additional RACH burst conf	
Number_preamble_scrambling_options[n]	3
}	
if (((3*N_RACH_burst_configurations+4) % 8) !=0) {	
reserved (length= 8-((3*N_RACH_burst_configurations+4)%8)	0
}	
UT_ID	16
GES_ID	8
// Comment : RTN channel configuration	
// Comment: Additional RTN channel config	
for (n=0; n < Number_RTN_channels; n++) {	
RTN_channel_configuration(N_supported_RACH_confs[n])	
Traffic_status	8
}	
// Comment: RTN Congestion Information	
// Comment: Additional CC config	
for (i=0 ; i < N_CC_parameter_sets; i++) {	
CC_config(N_CC_cong_states)	8

}	
// Comment : FWD link config parameters (traffic carrier)	
FWD_carrier_frequency	16
FWD_carrier_type	4
// Other config parameters	
generic_retransmission_number	4
old_channel_empty_timeout	8
L3_update_timeout	8
Connection_close_timeout	8
FLC_loss_timeout	8
No_received_data_timeout	8
if (opt_has_no_predefined_RACH_burst_config_flag == '1') {	
// Comment: Additional RACH burst configurations	
for (m=0; m < N_RACH_burst_configurations; m++) {	
RACH_ burst_configuration(Number_preamble_scrambling_options[m])	
}	
}	
// Control code (optional)	
if (opt_has_control_code_flag == '1') {	
control_code_with_FWD_carrier_id	128
}	
}	
Length (as encoded)	968

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x13.
- **opt_has_no_RACH_burst_config_flag** (1 bit): Flag to indicate the presence of optional *RACH burst configuration* information
- **opt_has_control_code_flag** (1 bit): Flag to indicate that optional control code is present in the message.
- **Number_RTN_channels** (8 bits): Number of RTN carriers (maximum 256 RTN carriers).

- **N_supported_RACH_confs[m]** (4 bits): Number of supported RACH burst configuration for each RTN channel (m).
- **N_CC_cong_states** (8 bits): Number of congestion control traffic states. Default value is 4.
- **N_CC_parameter_sets** (4 bits): Number of remaining congestion control parameter sets. It is the complement of the parameter sets that were defined in the LOGON TABLE.
- **N_RACH_burst_configurations** (4 bits): Number of additional RACH burst configuration parameter sets that are described in this message.
- **Number_preamble_scrambling_options[n]** (3 bits): Number of preamble and scrambling option defined for each additional RACH burst configuration (n) defined later in this message.
- **UT_ID** (16 bits): The L2 address assigned to the UT. This field will be used in the encapsulation header as source/destination address of the UT for subsequent signalling and data exchange (as in the LOGON VALIDATION REQUEST message).
- **GES_ID** (8 bits): The L2 address of the assigned GES. This field will be used in the encapsulation header as source/destination address of the GES for subsequent signalling and data exchange (as in the LOGON VALIDATION REQUEST message).
- **RTN_channel_configuration**(): Structure inserted for each RTN channel. Refer to the common structure definition in section 11.10. If a *RTN_channel_id* has already been defined in the LOGON_TABLE, then the information included in this message LOGON INITIAL ACCEPT prevails
- **Traffic_status** (8 bits): Refer to the CC_STATUS message (D018-COM-FUN-3720). If this field is set to b00000000, then the UT must wait for the reception of a CC_STATUS message in order to obtain the required traffic status information.
- **CC_config(N_CC_cong_states)**: Additional Congestion control configuration. Refer to the common structure definition in section 11.10.
- **FWD_carrier_type** (4 bits): Refer to LOGON TABLE message (D018-COM-FUN-3520).
- **FWD_carrier_frequency** (16 bits)**:** Refer to LOGON TABLE message (D018-COM-FUN-3520).
- **generic_retransmission_number** (4 bits): Generic Control Procedure Retransmission Number as defined in D018-COM-FUN-2480. Unsigned INTEGER (0..15).
- **old_channel_empty_timeout** (8 bits): Old Channel Empty Timeout. Time-out value (in units of seconds), used in the HO procedures defined in D018-COM-FUN-2761, D018-COM-FUN-2771 and D018-COM-FUN-2781. Unsigned integer (0..255).
- **L3_update_timeout** (8 bits): L3 Update Timeout. Time-out value (in units of seconds), used in the HO procedures defined in D018-COM-FUN-2761, D018-COM-FUN-2771 and D018-COM-FUN-2781. Unsigned integer (0..255).

- **Connection_close_timeout** (8 bits): Connection Close Timeout. Time-out value (in units of seconds), used in the HO procedures defined in D018-COM-FUN-2761, D018-COM-FUN-2771 and D018-COM-FUN-2781. Unsigned integer (0..255).
- **FLC_close_timeout** (8 bits): FLC_loss_timeout. Time-out value (in units of seconds) to consider the FLC lost, as defined in D018-COM-FUN-3290. Unsigned integer (0..255).
- **No_received_data_timeout** (8 bits): Time-out value (in units of minutes) to consider itself logged-off, as defined in D018-COM-FUN-2613. Unsigned integer (0..255).
- **RACH_burst_configuration():** Additional RACH Configuration. Refer to the common structure definition in section 11.10. One structure is used for each additional RACH configuration.
- **control_code_with_ FWD_carrier_id** (128 bits): Control code (optional). Refer to the CS guidelines regarding the use of this field.

In the table below, the following notation is used to indicate the flags:

- Rf : opt_has_no_predefined_RTN_carriers_flag
- Cf : opt_has_control_code_flag
- Nr_RACH_ids: Number_RACH_ids
- rtx_Number: generic_control_procedure_retransmission_number
- #CC_tr_states: N_CC_cong_states
- #CCsets : N_CC_parameter_sets
- ps_opts#0 : Number_preamble_scrambling_options [m=0]
- supRACHchRTN#0: N_supported_RACH_confs [n=0]

The encoding of the message shall follow the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x13)</td><td>ID</td><td rowspan="6">Optional & variable fields</td></tr>
<tr><td>2</td><td>Rf</td><td>Cf</td><td colspan="2">R</td><td colspan="4">Nr_RACH_ids</td><td>Options Flags</td></tr>
<tr><td>3</td><td colspan="8">Number_RTN_carriers</td><td rowspan="4">Length</td></tr>
<tr><td>4</td><td colspan="4">supRACHchRTN#0</td><td colspan="4">supRACHchRTN#1</td></tr>
<tr><td>5</td><td colspan="8">#CC_tr_states</td></tr>
<tr><td>6</td><td colspan="4">#CCsets</td><td colspan="3">ps_opts#0</td><td>ps_</td></tr>
</table>

7	opts#1	ps_opts#2	ps_opts#3		
8	UT_ID			General infos	General infos
9					
10	GES_ID				
11	RTN channel ID #0			ID	RTN Channel Config
12	RTN center frequency #0			Frequency	
13					
14	RTN Chan Type	RTN Chan Serv		Other ch. info	
15	RACH ID	reserved		RACH supp	
16	Preamble_scrambling_mask				
17	Traffic_status			Status	
18	RTN channel ID #1			ID	RTN Channel Config
19	RTN center frequency #1			Frequency	
20					
21	RTN Chan Type	RTN Chan Serv		Other ch. info	
22	RACH ID #0	reserved		RACH supp	
23	Preamble_scrambling_mask #0				
24	Traffic_status#0			Status	
25	RACH ID #1	reserved		RACH supp	
26	Preamble_scrambling_mask #1				
27	Traffic_status#1			Status	
28	CC_cat_id	CC_sf	reserved	CC Type	RTN congestion control information
29	tx_backoff [k=0]			CC Config#0	
30	persistence [k=0]				
31	retransmission_timeout [k=0]				
…	…			…	
38	tx_backoff [k=3]			CC Config#3	
39	persistence [k=3]				
40	retransmission_timeout [k=3]				
41	CC_cat_id	CC_sf	reserved	CC Type	RTN congestion control information
42	tx_backoff [k=0]			CC Config#0	
43	persistence [k=0]				

44	retransmission_timeout [k=0]			
…	…		…	
51	tx_backoff [k=3]		CC Config#3	
52	persistence [k=3]			
53	retransmission_timeout [k=3]			
54	FWD_carrier_frequency		FWD infos	FWD Carrier Info
55				
56	FWD_carr._type	rtx_Number	Configuration parameters	Configuration parameters
57	old_channel_empty_timeout			
58	L3_update_timeout			
59	Connection_close_timeout			
60	FLC_close_timeout			
61	No_received_data_timeout			
62	RACH conf ID	reserved	RACH burst conf #0	RACH info (Optional)
63	DCH_channelization_code			
64	ACH_channelization_code			
65	Preamble_scrambling sequence_options			
…				
97				
98	RACH conf ID	reserved	RACH burst conf #1	
99	DCH_channelization_code			
100	ACH_channelization_code			
101	Preamble Sequence			
102				
103	Scrambling Sequence			
104				
105				
106	control_code_with_FWD_carrier_id		Control Code	Options
…				
121				

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: NCC_ID
- Destination address: UT_ICAO_ID (24 bits address of the UT)

11.4.3 Logon Validation Request

D018-COM-FUN-3660

The Logon Validation Request shall comply with the following structure:

Syntax	No. of bits
LOGONvalidRequest_payload_structure() {	
SigMsgID (0x14)	8
opt_has_control_code_flag	1
opt_has_CSP_ids_flag	1
Reserved	2
if (opt_has_CSP_ids_flag == '1') {	
N_csp_ids	4
} else {	
Reserved	4
}	
Reserved	2
Administrative_parameters_UT_manufacturer_id	4
Number_Satellite_MAC_address	4
NCC_ID	8
UT_ICAO_ID	24
for (i=0; i < Number_Satellite_MAC_address; i++) {	
UT_Satellite_MAC_address [i]	48
}	
ROHC_supported_profile	8
Preferred_MODCOD	4
if (opt_has_CSP_ids_flag == '1') {	
for (j=0; j < N_csp_ids; j++) {	

preferred_CSP_id [j]	4
}	
if (N_csp_ids %2 ==1) {	
reserved	4
}	
}	
if (opt_has_control_code_flag == '1') {	
control_code_with_FWD_carrier_id	128
}	
} without Options	**116**
with All Options	**248**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x14.
- **opt_has_control_code_flag** (1 bit): Flag to indicate that a control code is present in the message. This field shall be set to 1 if the corresponding flag was equal to 1 in the received LOGON INITIAL ACCEPT message (D018-COM-FUN-3650), otherwise it shall be set to 0.
- **opt_has_CSP_ids_flag** (1 bit): Flag to identify the presence of CSP Id parameter list in the optional fields..
- **Administrative_parameters_UT_manufacturer_id** (4 bits): Refer to D018-COM-FUN-3640.
- **NCC_ID** (8 bits): This field must be set by the UT to the same value as the NCC_ID field included in the encapsulation header of the LOGON REQUEST message as L2 destination address.
- **UT_ICAO_ID** (24 bits): 24-bit ICAO address of the aircraft where the UT is installed. This is the value referred to in D018-COM-FUN-0430.
- **UT_Satellite _MAC_address** (48 bits): Refer to D018-COM-FUN-3640.
- **Number_Satellite_MAC_address** (4 bits)**:** Refer to D018-COM-FUN-3640.
- **ROHC_supported_profile** (8 bits): Refer to D018-COM-FUN-3640.
- **Preferred_MODCOD** (4 bits): Preferred FWD link MODCOD configuration. Refer to D018-COM-FUN-3750. If at the time of sending the LOGON VALIDATION REQUEST message is not known, then this field shall be set to b0000 (most robust MODCOD).
- **preferred_CSP_id** (4 bits): Preferred CSP Id.

- **control_code_with_FWD_carrier_id** (128 bits): This field shall only be present if the flag **opt_has_control_code_flag** is set to 1, otherwise it shall not be included. When present, this field shall have the same value as the equivalent field included in the received LOGON INITIAL ACCEPT message (D018-COM-FUN-3650).

In the table below, the following notation is used to indicate the flags:

- Cf : opt_has_control_code_flag
- CPf : opt_has_CSP_ids_flag
- adm_UT_manf_id: Administrative_parameters_UT_manufacturer_id
- N_MAC_addr.: Number_Satellite_MAC_address

The encoding of the message shall follow the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x14)</td><td>ID</td><td rowspan="3">Optional & variable fields</td></tr>
<tr><td>2</td><td>Cf</td><td>CPf</td><td colspan="2">reserved</td><td colspan="4">N_CSP_ids</td><td>Options Flags</td></tr>
<tr><td>3</td><td colspan="4">adm_UT_manf_id</td><td colspan="4">N_MAC_addr.</td><td>Length</td></tr>
<tr><td>4</td><td colspan="8">NCC_ID</td><td rowspan="12">General infos</td><td rowspan="12">General infos</td></tr>
<tr><td>5</td><td colspan="8" rowspan="3">UT_ICAO_address</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td></tr>
<tr><td>8</td><td colspan="8" rowspan="6">UT_satellite_ MAC_address</td></tr>
<tr><td>9</td></tr>
<tr><td>10</td></tr>
<tr><td>11</td></tr>
<tr><td>12</td></tr>
<tr><td>13</td></tr>
<tr><td>14</td><td colspan="8">ROHC_supported_profile</td></tr>
<tr><td>15</td><td colspan="4">Reserved</td><td colspan="4">Pref_MODCOD</td></tr>
<tr><td>16</td><td colspan="4">CSP_id_pref#1</td><td colspan="4">CSP_id_pref#2</td><td>CSP</td><td rowspan="4">Options</td></tr>
<tr><td>17</td><td colspan="8" rowspan="3">control_code_with_FWD_carrier_id</td><td rowspan="3">Control Code</td></tr>
<tr><td>…</td></tr>
<tr><td>32</td></tr>
</table>

The encapsulation header shall also include the following field(s) with the specified value:

- Length: corresponding to message options and variable length
- Source address: UT_ID
- Destination address: GES_ID

11.4.4 Logon Validation Accept

D018-COM-FUN-3670

The Logon Validation Accept shall comply with the following structure:

Syntax	No. of bits
LOGONvalidAccept_payload_structure() {	
SigMsgID (0x15)	8
opt_has_control_code_flag	1
GES_confirmation_required_flag	1
NCC_confirmation_required_flag	1
ROHC_used_flag	1
Include_destination_address_flag	1
reserved	7
Number_OSI_address	4
ARQ_support_per_flow_mask	16
rl_path_and_processing_time	8
GES_Satellite_MAC_address	48
for (i=0; i < Number_OSI_address; i++) {	
8208-AGR-Address [i]	48
CSP_associated [i]	4
}	
reserved	4
ROHC_supported_profile	8
if (opt_has_control_code_flag == '1') {	
control_code_for_NCC	128
}	
} without Options	**160**
with All Options	**288**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x15.
- **opt_has_control_code_flag** (1 bit): Flag to indicate the presence of the control code.

- **GES_confirmation_required_flag** (1 bit): Flag used for cases where the GES requests an additional confirmation message from the UT. This is the flag referred to as “GES confirm” in ST3 of D018-COM-FUN-2551. If set to one, the logon step 6 indicated in D018-COM-FUN-2550 will be performed.
- **NCC_confirmation_required_flag** (1 bit): Flag used to perform the logon validation directly with the NCC. This is the flag referred to as “NCC confirm” in ST3 of D018-COM-FUN-2551. If set to one, the logon steps 7 and 8 indicated in D018-COM-FUN-2550 will be performed.
- **ROHC_used_flag** (1 bit): Flag to indicate that ROHC will be used.
- **Include_destination_address_flag** (1 bit). Flag used to signal whether the RTN link bursts shall include the L2 destination address field, identifying the target GS element. If set to one, the AF field indicated in D018-COM-FUN-1780 can only be set to the values AF = {b00, b10, b11}. This flag is not applicable for the encapsulation of signalling messages.
- **ARQ_support_per_flow_mask** (16 bits): This parameter indicates whether a FWD link flow supports ARQ and therefore must be acknowledged by the UT following D018-COM-FUN-0760. This field is a mask: a bit in the position *i* (LSB first) is set to one if the flow with identifier *FID* = *i* has to be acknowledged. The parameter *FID* is defined in D018-COM-FUN-0830.
- **rl_path_and_processing_time** (8 bits): Value used by the UT to estimate the time at which transmitted RACH bursts are available at GS, as specified in D018-COM-FUN-1625. It specifies time expressed in ms units. It is encoded as an INTEGER (0..255) with a resolution of 10 ms to encode values in the range of [0 to 2.55] s.
- **ROHC_supported_profile** (8 bits): Set of ROHC profiles that can be used for communications with the GES. Refer to D018-COM-FUN-3640 for the coding of this field.
- **GES_Satellite_MAC_address** (48 bits): Satellite MAC address of the GES, as defined in D018-COM-FUN-0468.
- **8208-AGR-Address** (48 bits): OSI 8208 DTE address of the AGR within the GES. Refer to D018-COM-FUN-0680. The byte sequence of this field is encoded as big endian (first byte is the most significant). Several addresses are possible, with each of them associated with a different CSP.
- **CSP_associated** (4 bits): CSP identifier associated with the *8208-AGR-Address* parameter. This field is coded in the same way as the *Administrative_parameters_CSP_id* field (refer to D018-COM-FUN-3520).
- **Number_ OSI_ address** (4 bits): number of elements in the previous 2 fields (OSI address loop).
- **control_code_for_NCC** (128 bits): Control code (optional). Refer to the CS guidelines regarding the use of this field.

The encoding of the message shall follow the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x15)</td><td>ID</td><td rowspan="3">Optional & variable fields</td></tr>
<tr><td>2</td><td>Cf</td><td>Gf</td><td>Nf</td><td>Rf</td><td>Df</td><td colspan="3">reserved</td><td>Options Flags</td></tr>
<tr><td>3</td><td colspan="4">reserved</td><td colspan="4">N_DTE_addr</td><td>Length</td></tr>
<tr><td>4</td><td colspan="8" rowspan="2">ARQ_support_per_flow_mask</td><td rowspan="15">General infos</td><td rowspan="15">General infos</td></tr>
<tr><td>5</td></tr>
<tr><td>6</td><td colspan="8">rl_path_and_processing_time</td></tr>
<tr><td>7</td><td colspan="8" rowspan="6">GES_Satellite_MAC_address</td></tr>
<tr><td>8</td></tr>
<tr><td>9</td></tr>
<tr><td>10</td></tr>
<tr><td>11</td></tr>
<tr><td>12</td></tr>
<tr><td>13</td><td colspan="8" rowspan="3">8208-AGR-Address #0</td></tr>
<tr><td>…</td></tr>
<tr><td>18</td></tr>
<tr><td>19</td><td colspan="3">CSP_ass #0</td><td colspan="5">reserved</td></tr>
<tr><td>20</td><td colspan="8">ROHC_supported_profile</td></tr>
<tr><td>21</td><td colspan="8" rowspan="3">control_code
_for_NCC</td><td rowspan="3">Control Code</td><td rowspan="3">Options</td></tr>
<tr><td>…</td></tr>
<tr><td>36</td></tr>
</table>

In the table above, the following notation is used to indicate the flags:

- Cf : opt_has_control_code_flag
- Gf: GES_confirmation_required_flag
- Nf: NCC_confirmation_required_flag
- Rf: ROHC_used_flag
- Df: Include_destination_address_flag
- N_DTE_addr: Number_OSI_address

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: GES_ID
- Destination address: UT_ID

11.4.5 Logon Validation ACK

D018-COM-FUN-3680

The Logon Validation ACK shall comply with the following structure:

Syntax	No. of bits
LOGONvalidACK_payload_structure() {	
SigMsgID (0x16)	8
}	**8**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x16.

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x16)								ID	Opt. & var. Flds

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: UT_ID
- Destination address: GES_ID

11.4.6 Logon Accomplished

D018-COM-FUN-3690

The Logon Accomplished message shall comply with the following structure:

Syntax	No. of bits
LOGONaccomplished_payload_structure() {	
SigMsgID (0x18)	8
opt_has_control_code_flag	1
reserved	7
if (opt_has_control_code_flag == '1') {	
control_code_for_NCC	128
}	
} without Options	**16**
with All Options	**144**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x18.
- **opt_has_control_code_flag** (1 bit): Flag to indicate the presence of the control code. This field shall be set to 1 by the UT if the equivalent flag was equal to 1 in the received LOGON VALIDATION ACCEPT message (D018-COM-FUN-3670).
- **control_code_for_NCC** (128 bits): This field must be set by the UT to the same value as the equivalent field included in the received LOGON VALIDATION ACCEPT message (D018-COM-FUN-3670). This field shall only be present if included there, otherwise it shall not be included.

The encoding of the message shall follow the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x18)</td><td>ID</td><td rowspan="2">Optional & variable fields</td></tr>
<tr><td>2</td><td>Cf</td><td colspan="7">Reserved</td><td>flags/ length</td></tr>
<tr><td>3</td><td colspan="8" rowspan="3">control_code_for_NCC</td><td rowspan="3">Control Code</td><td rowspan="3">Options</td></tr>
<tr><td>…</td></tr>
<tr><td>18</td></tr>
</table>

In the table above, the following notation is used to indicate the flags:

- Cf : opt_has_control_code_flag

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: UT_ID
- Destination address: NCC_ID

11.4.7 Logon Accomplished ACK

D018-COM-FUN-3691

The Logon Accomplished ACK message shall comply with the following structure:

Syntax	No. of bits
LOGONaccomplishedACK_payload_structure() {	
SigMsgID (0x19)	8
}	**8**

The message content is built from the following information:

- **SigMsgID** (8 bits) to identify the signalling message and shall be equal to 0x19.

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x19)								ID	Opt. & var. Flds

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: NCC_ID
- Destination address: UT_ID

11.4.8 Logon Refused

D018-COM-FUN-3695

The syntax of this message shall conform to the following table:

Syntax	No. of bits
LOGONrefused_payload_structure() {	
SigMsgID (0x1F)	8

Logon_refusal_reason	8
}	**16**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x1F.
- **Logon_refusal_reason** (8 bits): An Id that describes the reason of logon refusal. A value of 0xFF is used if ground station does not provide additional reason for refusal. The reason shall be encoded using following table:

Logon refusal reason	Value
NCC logon refused	0x01
GES logon refused	0x02
RTN channel not usable	0x03
FWD channel not usable	0x04
Preference does not fit	0x05
Performance not achieved	0x06
Wrong control code	0x07
ICAO, @L2 not match	0x08
Registration not completed	0x09
Wrong NCC L2 address	0x0A
Wrong GES L2 address	0x0B
Wrong message Id	0x0C
No additional information	0xFF
Future use	Other values

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x1F)								ID	Opt. & var. Flds
2	Logon_refusal_reason								Reason	General infos

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: GES_ID or NCC_ID (depending on requester)
- Destination address: UT_ID

11.4.9 Logoff Request

D018-COM-FUN-3700

The Logoff request message shall comply with the following structure:

Syntax	No. of bits
LOGOFFrequest_payload_structure() {	
SigMsgID (0x1A)	8
opt_has_control_code_flag	1
reserved	7
if (opt_has_control_code_flag == '1') {	
control_code_for_GES	128
}	
} without Option	**16**
with Option	**144**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x1A.
- **opt_has_control_code_flag** (1 bit): Flag to indicate the presence of a control code (see below). This field only makes sense for ground-initiated logoff; it is set to 0 for UT-initiated logoff.
- **control_code_for_GES** (128 bits): Control code. This field must be set by the UT to the same value as the equivalent field included in the received LOGOFF REQUEST GROUND message (D018-COM-FUN-3713). This field is mandatory if included there, otherwise it shall not be included.

In the table below, the following notation is used to indicate the flags:

- Cf : opt_has_control_code_flag

The encoding of the message shall follow the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x1A)</td><td>ID</td><td rowspan="2">Optional & variable fields</td></tr>
<tr><td>2</td><td>Cf</td><td colspan="7">Reserved</td><td>flags/ length</td></tr>
<tr><td>3</td><td colspan="8" rowspan="3">control_code_for_GES</td><td rowspan="3">Control Code</td><td rowspan="3">Options</td></tr>
<tr><td>…</td></tr>
<tr><td>18</td></tr>
</table>

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: UT_ID
- Destination address: GES_ID

11.4.10 Logoff request ACK

D018-COM-FUN-3710

The Logoff request ACK message shall comply with the following structure:

Syntax	No. of bits
LOGOFFrequestACK_payload_structure() {	
MsgID (0x1B)	8
opt_has_control_code_flag	1
NCC_confirmation_required_flag	1
Reserved	6
NCC_ID	8
if (opt_has_control_code_flag == '1') {	
control_code_for_NCC	128
}	
} without Options	**24**
with All Options	**152**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x1B.
- **opt_has_control_code_flag** (1 bit): Flag to indicate the presence of the control code.
- **NCC_confirmation_required_flag** (1 bit): This flag indicates that step 3 as described in D018-COM-FUN-2600 must be performed and that a LOGOFF CONFIRM message (D018-COM-FUN-3711) must be sent to the NCC.
- **NCC_ID** (8 bits): The L2 address of the NCC currently associated with the UT, as defined in D018-COM-FUN-0450. This parameter is included in the Source/Destination address field of the encapsulation header in the (optional) LOGOFF CONFIRM / LOGOFF CONFIRM ACK messages (step 3 and step 4).
- **control_code_for_NCC** (128 bits): Control code (optional). Refer to the CS guidelines regarding the use of this field.

The encoding of the message shall follow the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x1B)</td><td>ID</td><td rowspan="2">Optional & variable fields</td></tr>
<tr><td>2</td><td>Cf</td><td>Nf</td><td colspan="6">reserved</td><td>flags/ length</td></tr>
<tr><td>3</td><td colspan="8">NCC_ID (optional)</td><td>Address</td><td>General infos</td></tr>
<tr><td>4</td><td colspan="8" rowspan="3">control_code_for_NCC</td><td rowspan="3">Control Code</td><td rowspan="3">Options</td></tr>
<tr><td>…</td></tr>
<tr><td>18</td></tr>
</table>

In the table above, the following notation is used to indicate the flags:

- Cf : opt_has_control_code_flag
- Nf: NCC_confirmation_required_flag

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: GES_ID
- Destination address : UT_ID

11.4.11 Logoff confirm

D018-COM-FUN-3711

The Logoff confirm message shall comply with the following structure:

Syntax	No. of bits
LOGOFFconfirm_payload_structure() {	
SigMsgID (0x1C)	8
opt_has_control_code_flag	1
reserved	7
GES_ID	8
if (opt_has_control_code_flag == '1') {	
control_code_for_NCC	128
}	
} without Options	**24**
with All Options	**152**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x1C.
- **opt_has_control_code_flag** (1 bit): Flag to indicate the presence of the control code.
- **GES_ID** (8 bits): The L2 address of the GES currently associated with the UT, as defined in D018-COM-FUN-0450.
- **control_code_for_NCC** (128 bits): This field must be set by the UT to the same value as the equivalent field included in the received LOGOFF REQUEST ACK message (D018-COM-FUN-3710). This field shall only be present if included there, otherwise it shall not be included.

The encoding of the message shall follow the format described below:

<table>
<tr><td></td><td colspan="8">Content in bits (MSB first)</td><td></td><td></td></tr>
<tr><td>Byte</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td></tr>
<tr><td>1</td><td colspan="8">SigMsgID (0x1C)</td><td>ID</td><td rowspan="2">Optional & variable fields</td></tr>
<tr><td>2</td><td>Cf</td><td colspan="7">reserved</td><td>flags/ length</td></tr>
</table>

3	GES_ID	General infos	General infos
4	control_code_for_NCC	Control Code	Options
…			
19			

In the table above, the following notation is used to indicate the flags:

- Cf : opt_has_control_code_flag

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: UT_ID
- Destination address: NCC_ID

11.4.12 Logoff confirm ACK

D018-COM-FUN-3712

The Logoff confirm ACK message shall comply with the following structure:

Syntax	No. of bits
LOGOFFconfirmACK_payload_structure() {	
SigMsgID (0x1D)	8
}	**8**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x1D.

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x1D)								ID	Opt. & var. Flds

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: NCC_ID
- Destination address: UT_ID

11.4.13 Logoff request - ground

D018-COM-FUN-3713

The Logoff request - ground message shall comply with the following structure:

Syntax	No. of bits
LOGOFFrequestGRD_payload_structure() {	
SigMsgID (0x1E)	8
opt_has_control_code_flag	1
reserved	7
GES_ID	8
if (opt_has_control_code_flag == '1') {	
control_code_for_GES	128
}	
} without Option	**24**
with Option	**152**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x1E.
- **GES_ID** (8 bits): The L2 address (as defined in D018-COM-FUN-0450) of the GES where the logoff shall be performed.
- **opt_has_control_code_flag** (1 bit): Flag to indicate the presence of the control code.
- **control_code_for_GES** (128 bits): Control code (optional). Refer to the CS guidelines regarding the use of this field.

In the table below, the following notation is used to indicate the flags:

- Cf : opt_has_control_code_flag

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x1E)								ID	Optional & variable fields
2	Cf	Reserved							Flags	
3	GES_ID								Address	General infos
4	control_code_for_GES								Control Code	Options
…										
19										

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: GES_ID or NCC_ID (depending on requester)
- Destination address: UT_ID

11.5 Radio Resource Management

The signalling structures used for radio resource management are related to RA congestion control functions.

11.5.1 RACH Congestion control

D018-COM-FUN-3720

The RACH Congestion control message CC_STATUS shall comply with the following structure:

Syntax	No. of bits
cc_status_payload_structure() {	
SigMsgID (0x51)	8
Num_channels	8
for (i=0; i < Num_channels; i++) {	
RTN_channel_id	8
Traffic_status	8
}	
}	16*Num_channels+16

The message content is built using the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x51.
- **Num_channels** (8 bits): Number of channel in the subsequent loop of reported channel status.
- **RTN_channel_id** (8 bits): Channel identifier. Value of the *RTN_channel_id* field included in the *RTN_channel_configuration* structure of the referenced RTN link channel. Refer to section 11.10 below.
- **Traffic_status** (8 bits): Measured traffic load of the channel (low, medium, high, congested), as used for RTN link congestion control (refer to D018-COM-FUN-3430). If the channel is disabled, then the UT shall not use the channel. This field is coded as indicated below:

Traffic Load Status	Value
Reserved	0b00000000
Low	0b00000001
Medium	0b00000010
High	0b00000011
Congested	0b00000100
Channel Disabled	0b00001111
Reserved	Any other value

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)							
Byte	8	7	6	5	4	3	2	1
1	SigMsgID (0x51)							
2	Num_channels							
3	Channel id (i=0)							
4	Traffic_status (i=0)							
5	Channel id (i=1)							
6	Traffic_status (i=1)							
7	Channel id (i=2)							
8	Traffic_status (i=2)							

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: GES_ID or NCC_ID
- Destination address: 0xFFFF (broadcast)

11.6 Network synchronisation

Requirements removed.

11.7 ACM

11.7.1 FWD Preferred MODCOD Request

D018-COM-FUN-3740

The FWD Preferred MODCOD Request message shall comply with the following structure:

Syntax	**No. of bits**
prefModcodReq_payload_structure() {	
SigMsgID (0x41)	8
}	**8**

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)							
Byte	8	7	6	5	4	3	2	1
1	SigMsgID (0x41)							

The message is built using the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x41.

The encapsulation shall also include the following field(s) with the specified value:

- Source address: GES_ID
- Destination address: UT_ID

11.7.2 FWD Preferred MODCOD Indication

D018-COM-FUN-3750

The FWD Preferred MODCOD Indication message shall comply with the following structure:

Syntax	No. of bits
prefModcodInd_payload_structure() {	
SigMsgID (0x42)	8
Preferred_MODCOD	4
Sequence_number	4
}	**16**

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)							
Byte	8	7	6	5	4	3	2	1
1	SigMsgID (0x42)							
2	Preferred_MODCOD				Sequence_number			

The message is built using the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x42.
- **Preferred_MODCOD** (4 bits): MODCOD preferred by the UT, as indicated in D018-COM-FUN-3350. The 4 bits are coded as shown in the table below:

Modcod	ID	Preferred_Modcod
QPSK ¼	0	0b0000
QPSK 1/3	1	0b0001
QPSK 1/2	2	0b0010
QPSK 2/3	3	0b0011
8-PSK ½	4	0b0100
8-PSK 2/3	5	0b0101
16-APSK 2/3	6	0b0110

- **Sequence_number** (4 bits): Binary encoding of a Sequence Number (increasing value 0 to 15, reset to 0 after 15). INTEGER (0..15). Its value shall be set according to D018-COM-FUN-3401.

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: UT_ID
- Destination address: GES_ID

11.7.3 FWD Preferred MODCOD Confirmation

D018-COM-FUN-3760

The FWD Preferred MODCOD Confirmation message shall comply with the following structure:

Syntax	No. of bits
prefModcodConf_payload_structure() {	
SigMsgID (0x43)	8
Sequence_number	4
reserved	4
}	**16**

The message is built using the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x43.
- **Sequence_number** (4 bits): Identifier of the acknowledged FWD PREFERRED MODCOD INDICATION message (see D018-COM-FUN-3750). Its value shall be set to the equivalent field included in the acknowledged message. INTEGER (0..15).

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)							
Byte	8	7	6	5	4	3	2	1
1	SigMsgID (0x43)							
2	reserved				Sequence_number			

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: GES_ID

- Destination address: UT_ID

11.8 Redundancy

11.8.1 OSI Reset

D018-COM-FUN-3770

The OSI Reset message shall comply with the following structure:

Syntax	**No. of bits**
REDUNDANCY_OSI_RESET_payload_structure() {	
SigMsgID (0x61)	8
GES_ID	8
Reset_counter	2
reserved	6
}	**24**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x61.
- **GES_ID** (8 bits). L2 address of the GES that requests an OSI reset. It identifies the link that requires an OSI reset (refer to D018-COM-FUN-3465).
- **Reset_counter** (2 bits): Reset counter value, as indicated in D018-COM-FUN-3463. It shall be incremented by one for each new OSI reset indication (i.e., it shall not be modified if the same OSI RESET message is retransmitted several times). INTEGER (0..3).

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x61)								ID	Opt. & var. fields
2	GES_ID								Address	Address
3	Reset_counter								Counter	General infos

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: GES_ID
- Destination address: 0xFFFF (broadcast)

11.8.2 Keepalive

11.8.2.1 Poll request

D018-COM-FUN-3771

The POLL_REQUEST message shall comply with the following structure:

Syntax	No. of bits
poll_request_payload_structure() {	
SigMsgID (0x71)	8
requestID	8
}	**16**

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x71.
- **requestID** (8 bits): Identifier of the request (refer to D018-COM-FUN-3466). Binary encoding of a sequence number (increasing value 0 to 15, reset to 0 after 15). INTEGER (0..15).

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x71)								ID	Opt. & var. Flds
2	requestID								Counter	Request ID

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: GES_ID or NCC_ID
- Destination address: UT_ID (only a unicast destination address is valid)

11.8.2.2 Poll Response

D018-COM-FUN-3772

The POLL_RESPONSE message shall comply with the following structure:

Syntax	No. of bits
poll_response_payload_structure() {	
SigMsgID (0x72)	8
receivedRequestID	8
	16

The message content is built from the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x72.
- **receivedRequestID** (8 bits): Identifier of the associated POLL_REQUEST message (refer to D018-COM-FUN-3466).

The encoding of the message shall follow the format described below:

	Content in bits (MSB first)									
Byte	8	7	6	5	4	3	2	1		
1	SigMsgID (0x72)								ID	Opt. & var. fields
2	receivedRequestID								Counter	Rx Request ID

The encapsulation header shall also include the following field(s) with the specified value:

- Source address: UT_ID
- Destination address: GES_ID or NCC_ID. It shall be set to the source address of the answered POLL_REQUEST message.

11.9 ARQ ACK signalling

D018-COM-FUN-3773

The ACK signalling message syntax shall conform to the following table:

Syntax	No. of bits
ACK_payload_structure() {	
SigMsgID (0x31 (FWD) or 0x32 (RTN))	8
Flow_identifier	4

Packet_count	4
Fragment_count	6
Reserved	2
}	**24**

The signalling message is built using the following information:

- **SigMsgID** (8 bits): To identify the signalling message and shall be equal to 0x31 (FWD ACK) or 0x32 (RTN ACK).
- **Flow identifier** field (4 bits): This field is a copy of the Flow Identifier of the ACKed LPDU (refer to D018-COM-FUN-0730 for RTN ACK and to D018-COM-FUN-1670 for FWD ACK).
- **Packet Count** field (4 bits): This field shall be a copy of the Packet Counter of the ACKed LPDU (refer to D018-COM-FUN-0730 for RTN ACK and to D018-COM-FUN-1670 for FWD ACK).
- **Fragment Count** field (6 bits): This field shall be a copy of the Fragment Counter of the ACKed LPDU (refer to D018-COM-FUN-0730 for RTN ACK and to D018-COM-FUN-1670 for FWD ACK).
- **RFU** (2 bits): Reserved for future use. It shall be set to b00.

The encoding of the message shall follow the format described below:

Byte	8	7	6	5	4	3	2	1
1	SigMsgID							
2	Flow_identifier				Packet_count			
3	R		Fragment_count					

Encapsulation headers for ARQ ACK signalling messages shall follow the same rules as for data traffic.

11.10 Common structures

This section describes a number of signalling structures that are common to several signalling messages defined in the previous sections.

11.10.1 RACH_burst_configuration

Syntax	No. of bits

RACH_burst_configuration (N_preamble_scrambling_opts) {	
RACH_config_id	4
reserved	4
DCH_channelization_code	8
ACH_channelization_code	8
Lower_bound_power_dB	8
// Preamble_scrambling_sequence_options	
for (i=0; i < N_preamble_scrambling_opts; i++) {	
Preamble_Scrambling_struct() [i]	33
}	
if ((N_preamble_scrambling_opts % 8) != 0) {	
reserved (length= 8-N_preamble_scrambling_opts%8)	7
}	
} Length when N_preamble_scrambling_opts=8	**296**
} Length when N_preamble_scrambling_opts=1	**72**

The RACH_burst_configuration must be called with the following input information:

- **Number_preamble_scrambling_options** (4bits): Number of elements *Preamble_Scrambling_struct* used in the RACH_burst_configuration. The maximum number of elements is 8.

The RACH_burst_configuration content is built using the following information:

- **RACH_config_id** (4 bits): Identification of the RACH burst configuration (defined in D018-COM-FUN-1820), as indicated below:

RACH_config_id	**Value**
$RACH\_CR_{160}\_SF_{16}\_DB_{512}$	b0001
$RACH\_CR_{160}\_SF_{4}\_DB_{2048}$	b0010
$RACH\_CR_{160}\_SF_{16}\_DB_{288}$	b0011
$RACH\_CR_{160}\_SF_{4}\_DB_{976}$	b0100
For future use	Any other value

- **DCH_channelization_code** (8 bits): DCH channelization code to be used for the RACH burst configuration, referred to as parameter k_d in D018-COM-FUN-2290.

- **ACH_channelization_code** (8 bits): ACH channelization code to be used for the RACH burst configuration, referred to as parameter k_a in D018-COM-FUN-2300.
- **Lower_bound_power_dB** (8 bits): Value used by the UT to explicitly randomize the transmission power as indicated in D018-COM-FUN-3295. It specifies a gain in the logarithmic domain expressed in dB. It is encoded as an INTEGER (-255..0) with a resolution of 0.1 dB to encode values in the range of [-25.5 to 0] dB.
- **Preamble_Scrambling_struct()** (33 bits): Different pairs of *Preamble_sequence_generator* and *Scrambling_sequence_generator* values are provided as options for the RACH burst configuration, to be selected by the UT. This structure shall comply with the following definition:

Syntax	No. of bits
Preamble_Scrambling_struct () {	
Preamble_sequence_generator	9
Scrambling_sequence_generator	24
}	**33**

The *Preamble_Scrambling_struct* content is built using the following information:

- **Preamble_sequence_generator** (9 bits): Preamble sequence generator to be used for the RACH burst configuration, referred to as parameter *n* in D018-COM-FUN-2360.
- **Scrambling_sequence_generator** (24 bits): Complex scrambling sequence generator to be used for the RACH burst configuration, referred to as parameter *n* in D018-COM-FUN-2320.

11.10.2 RTN_channel_configuration

Syntax	No. of bits
RTN_channel_configuration (N_supported_RACH_confs) {	
RTN_channel_id	8
RTN_channel_frequency	16
RTN_channel_type	4
RTN_channel_service_mask	4
for (j=0; j < N_supported_RACH_confs; j++) {	
RACH_config_id	4
preamble_scrambling_options_mask	8

}	
}	
if ((N_Supported_RACH_confs %2)!= 0) {	
reserved	4
}	
} Length when N_supported_RACH_confs =1 (minimum)	**48**
Length when N_supported_RACH_confs =5 (default)	**92**

The RTN_channel_configuration must be called with the following input information:

- **N_supported_RACH_confs** (4 bits): Number of RACH burst configuration used in the RTN channel configuration.

The RTN_channel_configuration content is built using the following information:

- **RTN_channel_id** (8 bits): Channel identifier. It is used to reference a specific channel (e.g., in the CC_STATUS message) and it shall be at least unique per mobile link beam. Unsigned integer.
- **RTN_channel_frequency** (16 bits): Central Frequency for each RTN link channel to be used (mobile link frequency). Specified as an offset in steps of 1kHz from the RTN_link_band_central_frequency parameter defined in the LOGON TABLE message (D018-COM-FUN-3520). Signed integer.
- **RTN_channel_type** (4 bits): RTN channel type. This value shall be set to 0 for the current version of the CS.
- **RTN_channel_service_mask** (4 bits): This parameter identifies for which data services the channel shall be used, according to the following table. This field is a mask, i.e., support for a specific service is signalled by setting a specific bit to one as indicated in the table below.

RTN_service_mask	Associated bit in the mask
Voice	b0001
Data	b0010
Signalling	b0100
For future use	b1000

- **RACH_config_id** (4 bits). Identifier of the supported RACH configuration. Refer to the same field included in the *RACH_burst_configuration* structure (section 11.10.1).
- **preamble_scrambling_options_mask** (8 bits): This parameter allows the system to indicate the applicable subset of the preamble and scrambling generator options defined

in the *RACH_burst_configurations* data structure. This field is a mask: a bit in position i (LSB first) shall be set to one if the option with index i is supported.

11.10.3 CC_config

Syntax	No. of bits
CC_config (N_CC_cong_states) {	
CC_category_id	**3**
CC_SF_id	**2**
reserved	**3**
for (k=0; k < N_CC_cong_states; k++) {	
// Comment: Loop over trafficState	
// Comment: if (k=0) { trafficState='Low' }	
// Comment: if (k=1) { trafficState='Medium' }	
// Comment: if (k=2) { trafficState='High' }	
// Comment: if (k=3) { trafficState='Congested' }	
tx_backoff	**8**
persistence	**8**
retransmission_timeout	**8**
}	
} Length when N_CC_cong_states = 4 (default)	**104**

The RTN_channel_configuration must be called with the following input information:

- **N_CC_cong_states** (8 bits): Number of congestion control traffic states. Default value is 4 (Low/Medium/High/Congested).

The CC_config content is built using the following information:

- **CC_category_id** (3 bits): Identifier of the applicable CC category associated with the CC parameter set defined in the loop below:

CC_category_id	Value
Normal	000
High	001
Signalling	010
Reserved	Rest

- **CC_SF_id** (2 bits): Identifier of the SF associated with the CC parameter set defined in the loop below:

CC_SF_id	Value
SF = 4	00
SF = 16	01
Reserved	Rest

The following CC configuration parameters are assigned for each of the traffic states indicated in D018-COM-FUN-3420, starting with the *low traffic* traffic state.

- **tx_backoff** (8 bits): Value of the tx_backoff parameter indicated in D018-COM-FUN-3410 and used by the RTN link CC mechanism. It specifies time expressed in ms units. It is encoded as an INTEGER (0..255) with a resolution of 50 ms to encode values in the range of [0 to 12.75] s.
- **persistence** (8 bits): Value of the persistence parameter indicated in D018-COM-FUN-3410 and used by the RTN link CC mechanism. It specifies a probability. It is encoded as an INTEGER (0..100) with a resolution of 0.01 to encode values in the range of [0 to 1].
- **retransmission_timeout** (8 bits): Value of the retransmission_timeout indicated in D018-COM-FUN-3410 and used by the RTN link CC and retransmission mechanisms. It specifies a time expressed in ms. It is encoded as an INTEGER (0..255) with a resolution of 50 ms to encode values in the range of [0 to 12.75] s.

12. LOW RATE WAVEFORM

12.1 User plane specification

12.1.1 User plane forward link specification

12.1.1.1 Link layer specification

Refer to 8.6.1.

12.1.1.2 Physical layer specification

The following section specifies the physical layer of the user plane FWD link at 16 kbaud.

12.1.1.2.1 Burst types

D018-COM-FUN-4000

The User plane FWD link using a low rate waveform shall support the following burst types:

- FCH burst

D018-COM-FUN-4010

The FCH burst transmitted using low rate waveform shall always use the following MODCOD:

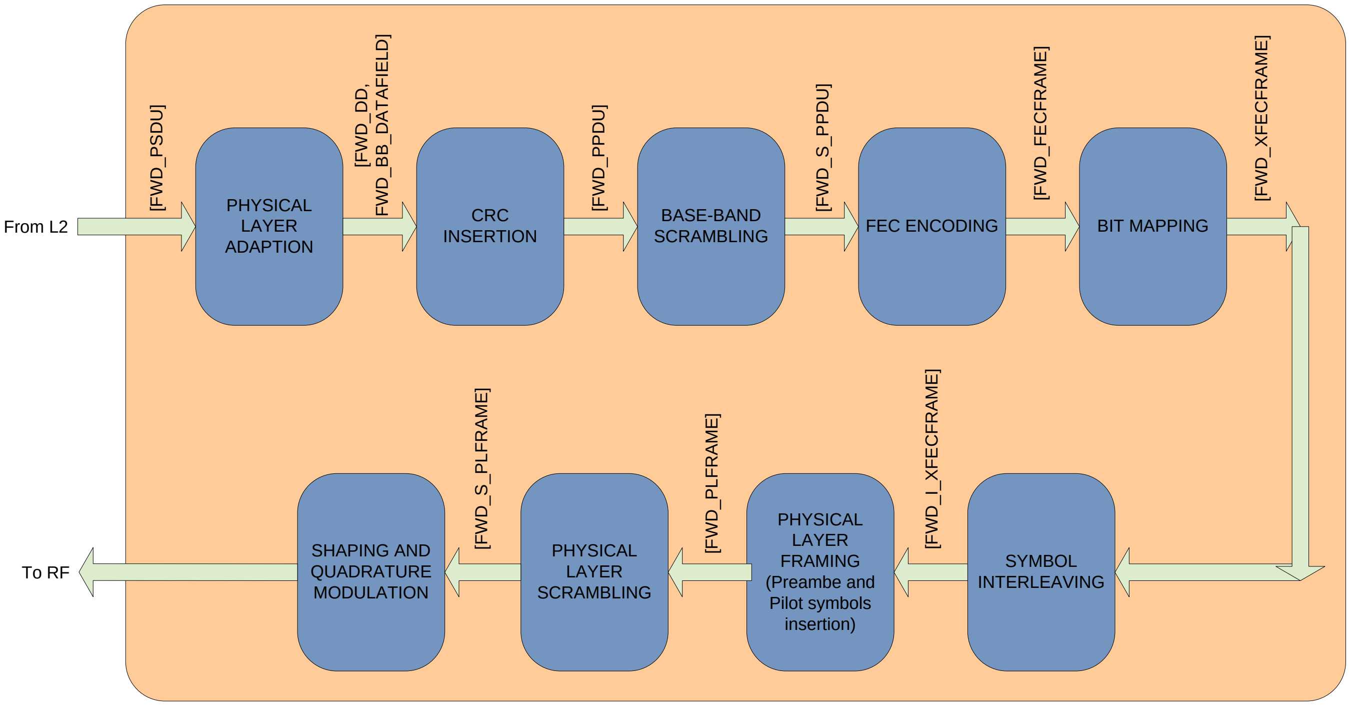
- QPSK 1/4

12.1.1.2.2 Burst waveform generation

D018-COM-FUN-4020

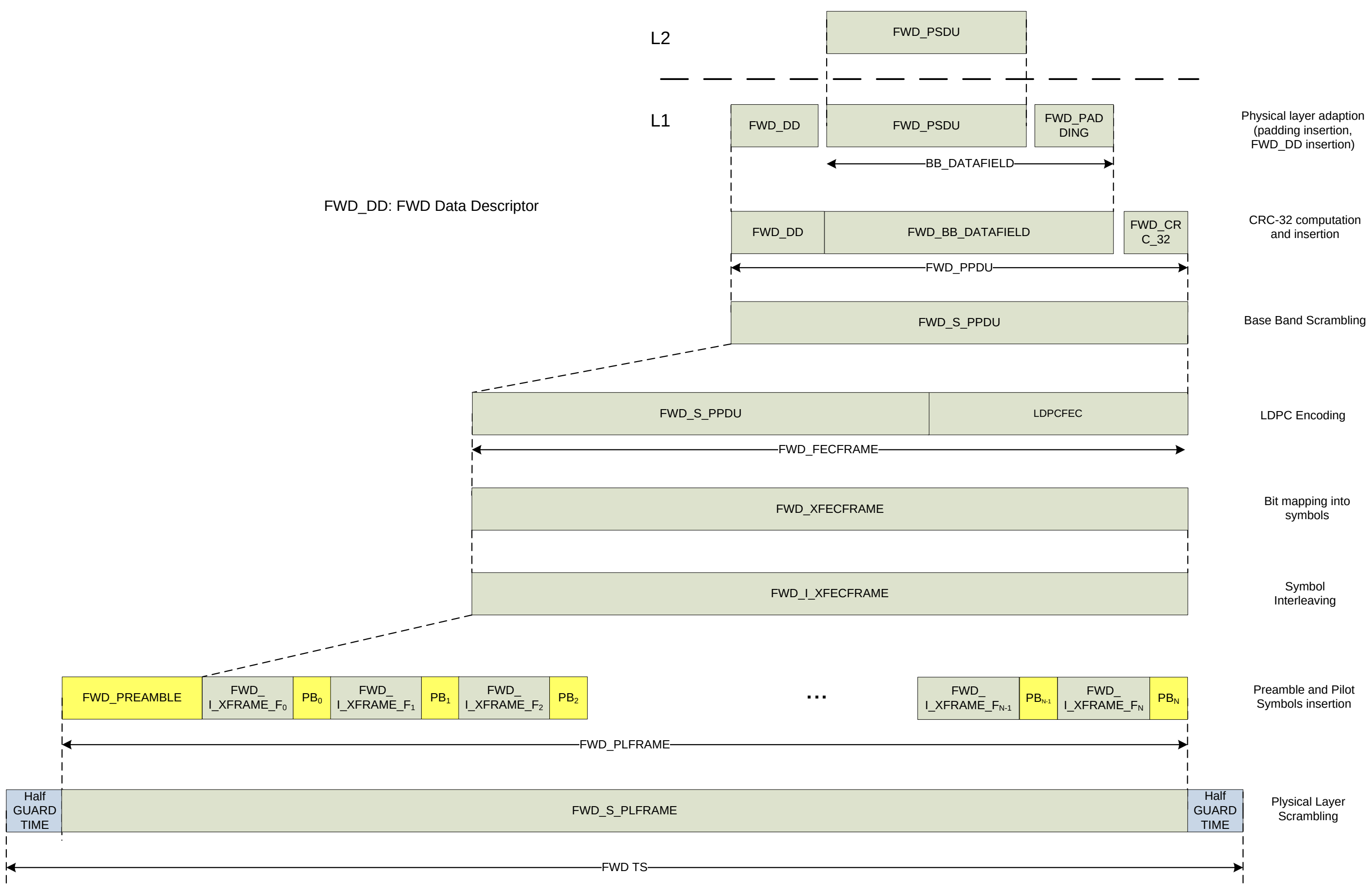
The low rate burst waveform generation shall be applied to the FWD_PSDU and composed of a sequence of functional modules as represented in the following figure. The functional modules are:

- Physical Layer Adaptation
- CRC insertion
- Base Band scrambling
- FEC Encoding
- Bit Mapping into Constellation
- Symbol Interleaving
- Physical Layer Framing
- Physical Layer Scrambling
- Base-band Pulse Shaping and Quadrature Modulation



D018-COM-FUN-4030

FCH bursts transmitted using low rate waveform shall follow the burst format presented in the following figure:



12.1.1.2.3 Physical Layer Adaption

D018-COM-FUN-4040

Physical Layer Adaption module for low rate waveform shall comply with requirement D018-COM-FUN 0880.

D018-COM-FUN-4050

Physical Layer Adaption module for low rate waveform shall comply with requirement D018-COM-FUN-0890.

D018-COM-FUN-4060

Physical Layer Adaption module for low rate waveform shall comply with requirement D018-COM-FUN-0900.

12.1.1.2.3.1 Interface with Layer 2

D018-COM-FUN-4070

The Physical Layer Adaptation module shall map FWD_PSDU of up to 4 types of layer 2 encapsulation protocols:

- FL Encapsulation, as defined in requirement D018-COM-FUN-0830
- Protocol 1 (RFU)
- Protocol 2 (RFU)
- Protocol 3 (RFU)

D018-COM-FUN-4080

The FWD_PSDU size (in bytes) shall be variable in the range. The maximum PSDU size (Max. N_{FWD_PSDU}) depends on the FWD_DD size (N_{FWD_DD}), as detailed in the following table:

Mode	**Maximum N_{FWD_PSDU} [Maximum PSDU size (bytes)]**	
	Non-extended FWD_DD (N_{FWD_DD} = 2 bytes)	**Extended FWD_DD (N_{FWD_DD} = 7 bytes)**
QPSK 1/4	186	181

Note: Maximum FWD PSDU size (Max. N_{FWD_PSDU}) corresponds with the FWD_BB_DATAFIELD size ($N_{FWD\_BB\_DFL}$)

12.1.1.2.3.2 Padding insertion

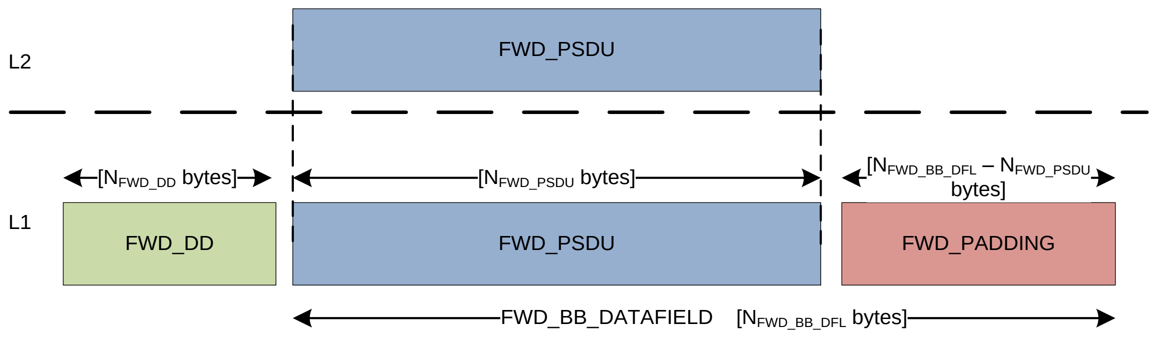
D018-COM-FUN-4090

Padding insertion module for low rate waveform shall comply with requirement D018-COM-FUN-0930.

12.1.1.2.3.3 FWD_DD (FWD Data Descriptor) insertion

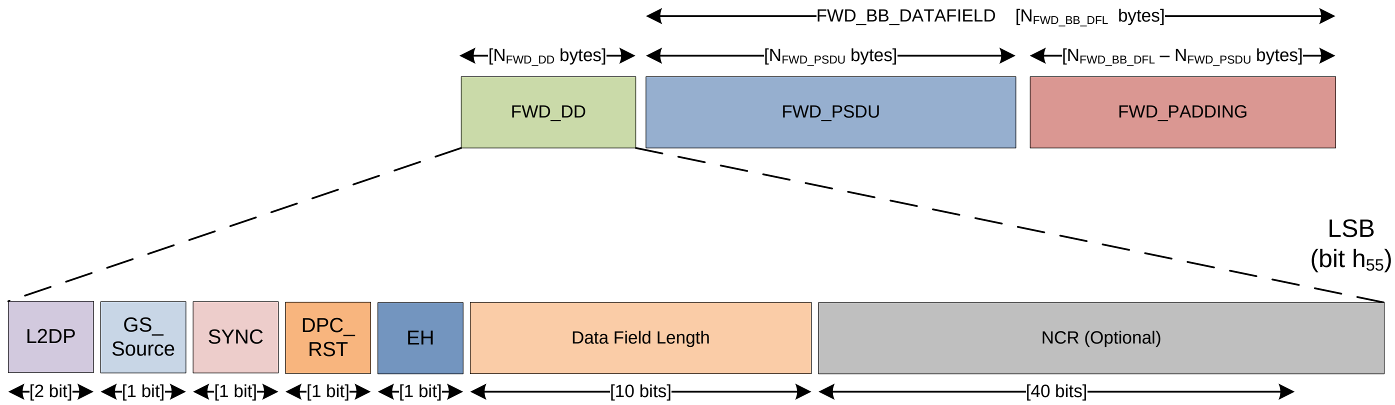
D018-COM-FUN-4100

A header (FWD_DD) of N_{FWD_DD}= 2 bytes (h =h_0, ..., h_{15}) - Non-extended FWD_DD - or N_{FWD_DD}= 7 bytes (h =h_0, ..., h_{55}) - Extended FWD_DD - shall be inserted in front of the FWD_BB_DATAFIELD according to the following figure:



The FWD_DD header contains the following fields:

Bits	Field	Field size	Description
h_0-h_1	L2DP	2 bits	It indicates the L2 Data Protocol (L2DP) used in the FWD_PSDU.
h_2	GS	1 bit	It indicates the GS Source (NCC or GES).
h_3	SYNC	1 bit	It indicates whether the burst is synchronous with respect to the NCC.
h_4	DPC_RST	1 bit	It indicates the need to reset the UT Doppler pre-compensation mechanism (this field is used in toggle mode).
h_5	EH	1 bit	It indicates whether the FWD_DD is extended or not.
h_6-h_{15}	DLF	10 bits	It contains the FWD_PSDU length (Data Length Field - DFL) in bytes. Range [0-1023].
h_{16}-h_{55}	NCR	40 bits	Optional field (only applicable for Extended FWD_DD). It carriers the value of the NCR.



D018-COM-FUN-4110

The contents of L2DP shall be as specified in requirement D018-COM-FUN-1010.

D018-COM-FUN-4120

The contents of GS field shall be as specified in requirement D018-COM-FUN-1020.

D018-COM-FUN-4130

The contents of SYNC field shall be as specified in requirement D018-COM-FUN-1030.

D018-COM-FUN-4131

The contents of the DPC_RST field (Doppler Pre-Compensation Reset) shall comply with requirement D018-COM-FUN-1031.

D018-COM-FUN-4132

The contents of the EH field (Extended Header) shall comply with requirement D018-COM-FUN-1033.

D018-COM-FUN-4133

The contents of the NCR field shall comply with requirement D018-COM-FUN-1034.

12.1.1.2.4 CRC insertion

D018-COM-FUN-4140

CRC insertion module for low rate waveform shall comply with requirement D018-COM-FUN-1040.

D018-COM-FUN-4150

CRC insertion module for low rate waveform shall comply with requirement D018-COM-FUN-1050.

D018-COM-FUN-4160

CRC insertion module for low rate waveform shall comply with requirement D018-COM-FUN-1060.

D018-COM-FUN-4170

CRC insertion module for low rate waveform shall comply with requirement D018-COM-FUN-1070.

D018-COM-FUN-4180

CRC insertion module for low rate waveform shall comply with requirement D018-COM-FUN-1080.

12.1.1.2.5 Base-Band Scrambling

D018-COM-FUN-4190

The Base-Band Scrambling module for the low rate waveform shall comply with requirement D018-COM-FUN-1090.

D018-COM-FUN-4200

The Base-Band Scrambling module for the low rate waveform shall comply with requirement D018-COM-FUN-1100.

D018-COM-FUN-4210

The Base-Band Scrambling module for the low rate waveform shall comply with requirement D018-COM-FUN-1110.

D018-COM-FUN-4220

The Base-Band Scrambling module for the low rate waveform shall comply with requirement D018-COM-FUN-1120.

D018-COM-FUN-4230

The Base-Band Scrambling module for the low rate waveform shall comply with requirement D018-COM-FUN-1130.

12.1.1.2.6 FEC Encoding (LDPC)

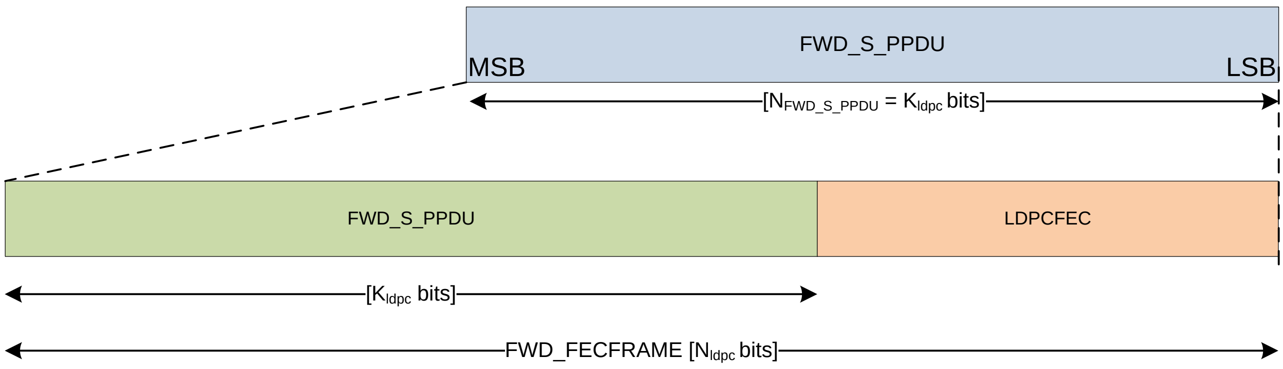
D018-COM-FUN-4240

The FEC Encoding module for low rate waveform shall perform the following function:

- Inner channel coding (IRA LDPC)

D018-COM-FUN-4250

The input stream of the FEC encoding module shall be a FWD_S_PPDU and the output stream a FWD_FECFRAME, as illustrated in the following figure:



D018-COM-FUN-4260

The FWD_S_PPDU shall be encoded using IRA LDPC code with code rate = 1/4 with the coding parameters defined in the following table:

MODCOD Id	MODCOD	$N_{FWD\_S\_PPDU}$ (bits) [K_{ldpc} (bits)]	$N_{FWD\_FECFRAME}$ (bits) [N_{ldpc} (bits)]
MODCOD0	QPSK 1/4	1536 bits	6144 bits

D018-COM-FUN-4270

The Inner channel coding for low rate waveform shall comply with requirement D018-COM-FUN-1180.

D018-COM-FUN-4280

The Inner channel coding for low rate waveform shall comply with requirement D018-COM-FUN-1190.

D018-COM-FUN-4290

The Inner channel coding for low rate waveform shall comply with requirement D018-COM-FUN-1200 for the specified MODCOD (QPSK 1/4).

12.1.1.2.7 Bit mapping into constellation

D018-COM-FUN-4300

The Bit Mapping module shall comply with requirement D018-COM-FUN-1250 for the MODCOD specified for the low rate waveform (QPSK 1/4, η_{MOD}=2).

D018-COM-FUN-4310

The Bit Mapping module shall comply with requirement D018-COM-FUN-1260 for the MODCOD specified for the low rate waveform (QPSK 1/4, η_{MOD}=2).

D018-COM-FUN-4320

The Bit Mapping module shall comply with requirement D018-COM-FUN-1270 for the MODCOD specified for the low rate waveform (QPSK 1/4, η_{MOD}=2).

D018-COM-FUN-4330

The Bit Mapping module shall comply with requirement D018-COM-FUN-1280 for the MODCOD specified for the low rate waveform (QPSK 1/4, η_{MOD}=2).

D018-COM-FUN-4340

The Bit Mapping module shall comply with requirement D018-COM-FUN-1290 for the MODCOD specified for the low rate waveform (QPSK 1/4, η_{MOD}=2).

D018-COM-FUN-4350

The Bit Mapping module shall comply with requirement D018-COM-FUN-1300 for the MODCOD specified for the low rate waveform (QPSK 1/4, η_{MOD}=2).

12.1.1.2.8 Symbol interleaving

D018-COM-FUN-4360

The Symbol Interleaving module shall interleave the symbols of 1 FWD_XFECFRAME in order to exploit the time diversity.

D018-COM-FUN-4370

The input stream of the Symbol Interleaving module shall be a FWD_XFECFRAME and the output a FWD_I_XFECFRAME.

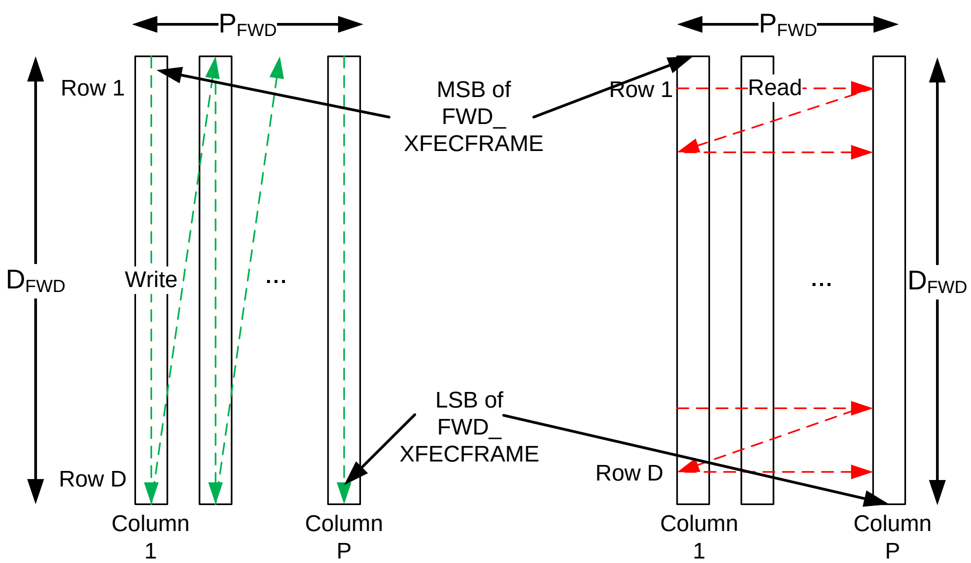
D018-COM-FUN-4380

One FWD_XFECFRAME shall be symbol interleaved using a block interleaver (row/column) with the parameters defined in the following table:

MODCOD Id	MODCOD	D_{FWD} (Number of Rows)	P_{FWD} (Number of Columns)
MODCOD0	QPSK 1/4	96	32

D018-COM-FUN-4390

At the transmitter end, the symbols shall be serially written in column-wise (starting from the MSB of FWD_XFECFRAME and ending with the LSB of FWD_XFECFRAME) and serially read out row-wise according to the following figure.



D018-COM-FUN-4400

At the receiver end, the symbols shall be serially written in row-wise and serially read out column-wise.

12.1.1.2.9 Physical Layer Framing

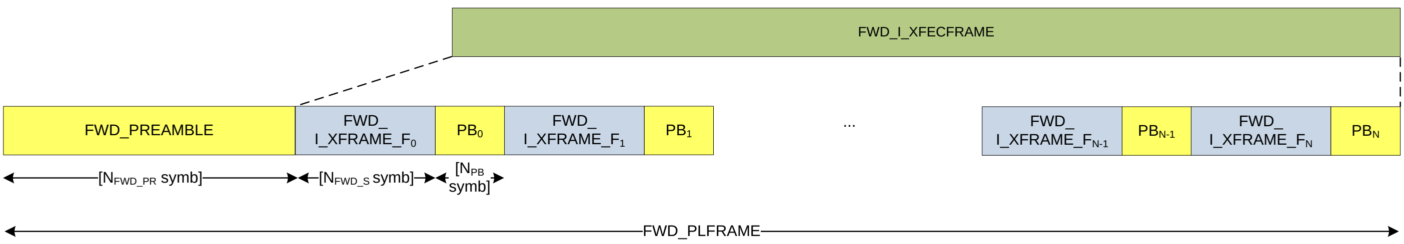
D018-COM-FUN-4410

The Physical Layer Framing module for low rate waveform shall perform the following process:

- PB (Pilot Block) insertion
- FWD_PREAMBLE insertion

D018-COM-FUN-4420

The input stream of the Physical Layer Framing sub-system shall be a FWD_I_XFECFRAME and the output a FWD_PLFRAME, as represented in the following figure.



12.1.1.2.9.1 PB (Pilot Blocks) insertion

D018-COM-FUN-4430

A Pilot Block (PB) shall be composed of N_{FWD_PB} = 6 un-modulated symbols with equal In-phase (I) and Quadrature (Q) components:

$$I_j = Q_j = \frac{\rho}{\sqrt{2}}, \qquad j = 0, \dots, N_{FWD\_PB} - 1$$

D018-COM-FUN-4440

PB shall be uniformly inserted within the FWD_I_XFECFRAME according to the following rule:

1. The FWD_I_XFRAME is divided into fragments called FWD_I_XFRAME_F_i (see requirement D018-COM-FUN-4420), resulting in:
   - 56 fragments of $N_{FWD\_I\_XFRAME\_F}$ = 54 symbols (FWD_I_XFRAME_F_0, ..., FWD_I_XFRAME_F_{55})
   - 1 last fragment of $N_{FWD\_I\_XFRAME\_FN}$ = 48 symbols (FWD_I_XFRAME_F_{56})
2. A PB is appended after each FWD_I_XFRAME_F_i, as illustrated in requirement D018-COM-FUN-4240

12.1.1.2.9.2 Preamble insertion

D018-COM-FUN-4450

The FWD_PREAMBLE shall consist of a $N_{FWD\_PREAMBLE}$ = 160 un-modulated symbols with equal In-phase (I) and Quadrature (Q) components:

$$I_k = Q_k = \frac{\rho}{\sqrt{2}}, \quad k = 0, \ldots, N_{FWD\_PREAMBLE} - 1$$

D018-COM-FUN-4460

The FWD_PREAMBLE shall be inserted before the first fragment of FWD_I_XFRAME (FWD_I_XFRAME_F_0), as illustrated in requirement D018-COM-FUN-4240.

12.1.1.2.10 Physical layer scrambling

D018-COM-FUN-4470

The Physical layer scrambling module for low rate waveform shall comply with requirement D018-COM-FUN-1520.

D018-COM-FUN-4480

The Physical layer scrambling module for low rate waveform shall comply with requirement D018-COM-FUN-1530.

D018-COM-FUN-4490

The Physical layer scrambling module for low rate waveform shall comply with requirement D018-COM-FUN-1540.

D018-COM-FUN-4500

The Physical layer scrambling module for low rate waveform shall comply with requirement D018-COM-FUN-1550.

(Note: R_n sequence shall be truncated to the FWD_PLFRAME size of the low rate waveform. i.e., $N_{FWD\_PLFRAME}$ = 3574 symbols)

12.1.1.2.11 Base-Band pulse shaping and quadrature modulation

D018-COM-FUN-4510

The Base-band pulse shaping and quadrature modulation module shall comply with requirement D018-COM-FUN-1560.

D018-COM-FUN-4520

The Base-band pulse shaping and quadrature modulation module shall comply with requirement D018-COM-FUN-1570.

D018-COM-FUN-4530

The Base-band pulse shaping and quadrature modulation module shall comply with requirement D018-COM-FUN-1580.

D018-COM-FUN-4540

The Base-band pulse shaping and quadrature modulation module shall comply with requirement D018-COM-FUN-1590.

12.2 Control plane specification

12.2.1 Control plane forward link specification

12.2.1.1 Link layer specification

Refer to 8.6.1

12.2.1.2 Physical layer specification

D018-COM-FUN-4550

Low rate carriers FWD Link Control Plane shall comply with the low rate waveform FWD Link User Plane specification detailed in section 12.1.1.2 above.

13. APPENDIX B: ADDRESSES OF PARITY BIT ACCUMULATORS FOR IRA LDPC

13.1 Addresses of parity bit accumulator for r=1/4 and k_{ldpc} = 1536 bits

145 925 1446 2183 2568
134 306 362 979 2245
147 158 2907 4447 4580
1316 2205 2920 3592 3795
340 2242 2441 2525 2757
2182 2580 2999 3725 4140
414 754 1166 2026 2668
127 2201 2567 3903 4031
1420 1604 1980 1986 3090
849 1909 2347 3365 4279
1196 2578 3170 3560 3684
2255 2487 3253 3645 4317

Table 13-1: Addresses of parity bit accumulator for r=1/4 and k_{ldpc} = 1536 bits

13.2 Addresses of parity bit accumulator for r=1/3 and k_{ldpc} = 2048 bits

469 962 2940 3752
2173 2953 3587 3894
202 1796 2110 4023
299 997 2431 3568
536 550 1676 2513
313 711 2157 3730
115 1710 2074 3040
463 673 2011 3828
1446 2093 2396 2839
592 846 1021 2279
672 1150 1873 4015
1522 2184 2687 3649
184 809 3234 3859
842 1716 3171 4057
171 533 1380 2234
172 1659 1974 2373

Table 13-2: Addresses of parity bit accumulator for r=1/3 and k_{ldpc} = 2048 bits

13.3 Addresses of parity bit accumulator for r=1/2 and k_{ldpc} = 3072 bits

1001 1397 1561 2604 2768
407 439 1589 2293 2416
37 828 1203 1376 1702
622 1027 1224 1845 2457
963 1188 1639 2109 2521
471 2215 2266 2372 2730
257 1120 1931 2103 2136
31 506 754 945 1811
449 695 739 1677 1946
451 854 1114 2934 2992
12 740 1719 1857 2022
854 2139 2274 2790 2805
418 1049 1777 2163 2672
443 1088 2286 2442 2878
97 1113 1720 2404 2682
657 1245 2174 2339 2837
1235 1602 1652 2308 2519
412 595 782 1481 1704
288 293 301 1419 1484
266 1752 2639 2724 3062
366 410 1096 1405 1484
697 1339 2247 2524 2725
242 1102 1311 2143 2716
646 944 1223 1426 1493

Table 13-3: Addresses of parity bit accumulator for r=1/2 and k_{ldpc} = 3072 bits

13.4 Addresses of parity bit accumulator r=2/3 and k_{ldpc} = 4096 bits

191 918 1451 1730
28 326 627 1272
168 439 595 1516
1696 1751 1901 1961
89 1325 1572 1712
510 1377 1396 1546
1073 1125 1694 1802
53 107 466 783
254 307 1337 1862
1206 1567 1610 1763
234 416 1175 1951
60 768 795 967

299 1068 1108 1553
61 273 1428 2008
226 285 488 1045
146 453 1086 1833
465 1097 1175 1948
215 1194 1282 1740
125 266 306 1876
35 941 1732 1947
363 1603 1694 1813
149 152 304 702
552 758 831 1024
241 1110 1449 1567
547 1245 1562 1861
1126 1982 2027 2032
112 235 822 1358
481 600 847 1911
127 215 1384 1617
1266 1481 1892 1948
274 329 588 1316
405 861 1091 1418

Table 13-4: Addresses of parity bit accumulator for r=2/3 and k_{ldpc} = 4096 bits

13.5 Addresses of parity bit accumulator r=1/2 and k_{ldpc} = 4608 bits

1821 2669 2747 3099 3393
1273 2195 2562 4062 4107
1 644 862 1355 1663
462 1468 2539 2977 3683
1387 2741 2824 3070 3734
31 591 906 2503 3924
896 1136 1227 1780 2547
485 872 3412 3898 3980
311 1532 1547 3962 4547
60 1178 1930 2787 3719
529 561 1563 2730 4106
312 2966 3684 3826 4495
2564 2797 2953 3180 3726
265 1097 1611 3165 3409
1166 1372 3625 3698 4247
44 1245 1759 2337 3832

1264 1630 2319 2461 3421
481 776 2305 2817 3190
226 1434 2215 2608 3994
248 3059 3370 3510 4037
1347 1787 2348 3159 3641
196 576 2000 4420 4607
605 1380 3422 3960 4382
281 877 2507 4135 4338
806 1515 1746 2229 2942
1762 2057 2933 3010 4243
405 1756 2954 4470 4497
1123 2457 2514 3974 4229
543 2544 2770 4457 4472
19 794 1004 2004 4517
1287 1922 2369 2377 2598
2526 2912 3964 4277 4521
94 244 3274 3316 4476
765 1569 2070 3967 4296
36 1951 2157 2927 4487
459 1644 1764 2059 3784

Table 13-5: Addresses of parity bit accumulator for r=1/2 and kldpc = 4608 bits

13.6 Addresses of parity bit accumulator r=2/3 and k_{ldpc} = 6144 bits

1400 1564 2615 2773
446 989 1593 2298
399 672 1219 2410
40 827 1369 1700
1025 1226 2454 2541
963 1198 1639 1788
263 472 2090 2263
185 507 944 3066
1356 1497 2155 2764
1066 1397 2029 2252
1392 1526 1619 2253
687 1662 2545 2830
574 2747 2797 2954
182 1035 1535 2694
220 1098 2503 2871
440 1963 2069 3016
2040 2225 2564 2673

490 1173 2172 2977
417 541 1196 1417
1082 1094 2733 2938
238 255 747 1331
1703 1924 2152 2718
173 209 631 1194
504 948 1520 2851
453 1827 2147 2244
1822 2029 2430 2956
974 1133 2311 2399
320 1146 3015 3024
619 1504 1681 3009
730 1145 2666 2876
990 1048 1847 3051
127 412 426 1121
1304 1421 1675 2580
609 648 1628 2821
405 806 1345 2770
303 770 1318 2699
569 652 656 2202
365 1089 1404 2083
476 1117 2338 2712
73 717 1454 2867
118 1239 1566 1898
1720 1827 2335 2783
825 1250 2293 2300
734 2098 2181 2283
803 2758 2991 3028
280 342 2549 2615
1553 1962 2568 2599
636 2360 2545 2995

Table 13-6: Addresses of parity bit accumulator for r=2/3 and k_{ldpc} = 6144 bits

13.7 Addresses of parity bit accumulator r=2/3 and k_{ldpc} = 8192 bits

763 2006 2831 3202
1772 2035 2424 3062
359 2867 3640 3676
32 781 2359 2809
512 900 905 2237

1822 2436 2538 3329
990 1649 1738 2869
351 1138 3013 3531
403 1318 1902 4041
618 675 1398 3071
192 1495 3695 3994
155 800 1143 2172
321 2276 3099 4092
1633 2932 2952 2973
61 1848 2533 3394
85 649 2542 2738
1623 1793 1900 2137
82 2332 2586 3943
957 1780 1978 2594
811 2669 3811 4004
963 2565 3259 3982
992 1550 1560 1893
159 1112 2310 3264
166 1103 1785 3905
547 1757 3157 3658
1488 1782 1806 3035
1451 1456 1470 2982
15 2552 2855 4001
727 2577 2696 2879
18 2132 2345 3564
1385 2306 2892 3940
2021 2451 2618 2813

Table 13-7: Addresses of parity bit accumulator for r=2/3 and k_{ldpc} = 8192 bits

14. APPENDIX C: REQUIREMENTS CHANGE RECORD

From v1.0 to v1.1:

PUID	Change Record
D018-COM-FUN-0260	Corrected typo: "used for" instead of "unicast".
D018-COM-FUN-0543	Removed as IPv6 does not have broadcast destination addresses.
D018-COM-FUN-0547	Removed reference to broadcast from requirement, as IPv6 does not have a broadcast destination address.
D018-COM-FUN-0730	Correction: Reference D018-COM-FUN-3780 should be D018-COM-FUN-3773. Placed "R" bits before the "FC" field to be aligned with D018-COM-FUN-3773.
D018-COM-FUN-0740	Explicitly state that the P/F field is not present in LPDUs belonging to a single-fragment NPDU. Minor wording changes.
D018-COM-FUN-0750	Changes in the SDL to replace "SET (ET_Timer)" by "Initialize ET_Timer". Corrected error in the definition of nReTx: "change" replaced by "chance". Removed the last paragraph in the description of the “ReTx_Timer_Timeout" event as it was considered potentially confusing: The SDL and its description is already covering what was stated in the paragraph. Typos corrected.
D018-COM-FUN-0760	Wording changes in the definition of fAckMark to make the description clearer. Wording changes in the description of the meaning of "acknowledged". Minor wording changes.
D018-COM-FUN-0780	Clarification: added note with pointer to requirements indicating payload size.
D018-COM-FUN-0830	Corrected wrong reference to "packet counter" in fragment count description. It is clarified that the CRC-32 is computed as indicated in D018-COM-FUN-1070 and D018-COM-FUN-1080, considering the PNPDU as input stream (this was now only included in D023). Replaced reference to 8192 bytes in Total Length field description by 8191 bytes, which is the correct maximum value.
D018-COM-FUN-0870	Figure updated (typo error): FWD_BB_DATAFIELD instead of BB_DATAFIELD.
D018-COM-FUN-1000	Figure updated (typo errors in the field names).
D018-COM-FUN-1032	Typo: "signal" instead of "signalling".

D018-COM-FUN-1530	Requirement updated (typo error in the formula).
D018-COM-FUN-1615	Correction. Corrected reference from D018-COM-FUN-0581 to D018-COM-FUN-0580.
D018-COM-FUN-1625	Correction. Corrected reference from D018-COM-FUN-0581 to D018-COM-FUN-0580.
D018-COM-FUN-1640	Minor wording change: "LPDUs, which are known" replaced by "LPDUs, that are known".
D018-COM-FUN-1660	Clarified location of reserved bits, now aligned with ARQ ACK definition (also regarding P/F field). Added clarification regarding bit 0 (MSB) of the Flow ID field, which shall be set to zero in the case of fragmented packets.
D018-COM-FUN-1690	Typo corrected.
D018-COM-FUN-1780	Corrected wrong section references. Now the applicable requirement (D018-COM-FUN-1660) is referenced directly. Added clarification on sequence number: This number is a cyclic counter of fragmented packets. Clarified that the CRC-32 is computed as indicated in D018-COM-FUN-1070 and D018-COM-FUN-1080, considering the PNPDU as input stream. Replaced reference to 4096 bytes in Total Length field description by 4095 bytes. Misleading statement "Otherwise, its presence depends on the value of the ARQ flag" has been removed from the FID field description. Misleading statement "Thus only the Packet count fields are present within this ARQ field" has been removed.
D018-COM-FUN-1890	Requirement updated (typo in the field names corrected).
D018-COM-FUN-1900	Requirement updated (typo errors corrected).
D018-COM-FUN-2050	Requirement updated (editorial error corrected)
D018-COM-FUN-2360	Requirement updated: - clarification added regarding the range of index n - clarification added regarding the range of index i for the generation of sequence x, y and zn
D018-COM-FUN-2551	Clarifications added: - Added note "Note: Carrier Id refers to the UT_known_RACH_configurations_mask field" - Added note (last bullet), defining n_max value (= generic_retransmission_number). - "Other messages" replaced by "Other logon messages" in the SDL diagrams. Correction: - In LOGON VALIDATION STATE, arrow coming from « NCC confirm = 1 » « YES » now enters the initialization block from

	above.
D018-COM-FUN-2610	Added missing LOGOFF REQUEST GROUND message to figure (first message, from the GS to UT). Description was already ok.
D018-COM-FUN-3460	Correction: removed CC_CONFIG from req 3460, as it has not been defined as stand-alone message.
D018-COM-FUN-3520	Minor corrections: added some missing encoding flags. Added required parameter N_CC_cong_states to CC_config() structure (in description). Added signalling message size. For entry RTN_channel_definition_per_NCC_flag, "above" has been replaced by "below".
D018-COM-FUN-3540	Added some missing encoding flags. Updated syntax table, as it had not been updated in the previous version. Correction:« reserved » field above HO_type has been set to 4 bits.
D018-COM-FUN-3592	Corrected signalling message sizes.
D018-COM-FUN-3640	Added missing table for ROHC_supported_profile filed.
D018-COM-FUN-3650	Minor corrections: added some missing encoding flags. Added required parameter N_CC_cong_states to CC_config() structure (in description). Added signalling message size.
D018-COM-FUN-3660	Correction: added 2 missing bits (reserved bits) in the syntax table.
D018-COM-FUN-3695	Added missing requirement label.
D018-COM-FUN-3770	Correction. Corrected reference from D018-COM-FUN-3464 to D018-COM-FUN-3465.

Table 14-1: Requirements Change Record from v1.0 to v1.1

END OF DOCUMENT

DISSEMINATION RIGHTS

Doc ID	Doc Title	Public	SESAR Stakeholders and ESA National Delegates	ANTARES participants	Iris Programme	ESA and Prime
D018	Communication Standard Technical Specifications	x	x	X	x	x