ETSI TR 101 545-4 V1.1.1 (2014-04)

Technical Report

Digital Video Broadcasting (DVB);
Second Generation DVB
Interactive Satellite System (DVB-RCS2);
Part 4: Guidelines for Implementation and Use of EN 301 545-2

Reference

DTR/JTC-DVB-324-4

Keywords

DVB, interaction, satellite

ETSI

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

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

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

Important notice

Copyright Notification

Contents

Intellectual Property Rights

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

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

Foreword

This Technical Report (TR) has been produced by Joint Technical Committee (JTC) Broadcast of the European Broadcasting Union (EBU), Comité Européen de Normalisation ELECtrotechnique (CENELEC) and the European Telecommunications Standards Institute (ETSI).

NOTE: The EBU/ETSI JTC Broadcast was established in 1990 to co-ordinate the drafting of standards in the specific field of broadcasting and related fields. Since 1995 the JTC Broadcast became a tripartite body by including in the Memorandum of Understanding also CENELEC, which is responsible for the standardization of radio and television receivers. The EBU is a professional association of broadcasting organizations whose work includes the co-ordination of its members' activities in the technical, legal, programme-making and programme-exchange domains. The EBU has active members in about 60 countries in the European broadcasting area; its headquarters is in Geneva.

European Broadcasting Union
CH-1218 GRAND SACONNEX (Geneva)
Switzerland
Tel: +41 22 717 21 11
Fax: +41 22 717 24 81

The Digital Video Broadcasting Project (DVB) is an industry-led consortium of broadcasters, manufacturers, network operators, software developers, regulatory bodies, content owners and others committed to designing global standards for the delivery of digital television and data services. DVB fosters market driven solutions that meet the needs and economic circumstances of broadcast industry stakeholders and consumers. DVB standards cover all aspects of digital television from transmission through interfacing, conditional access and interactivity for digital video, audio and data. The consortium came together in 1993 to provide global standardization, interoperability and future proof specifications.

The present document is part 4 of a multi-part deliverable covering the Second Generation DVB Interactive Satellite System (DVB-RCS2), as identified below:

TS 101 545-1: "Overview and System Level specification";

EN 301 545-2: "Lower Layers for Satellite standard";

TS 101 545-3: "Higher Layers Satellite Specification";

TR 101 545-4: "Guidelines for Implementation and Use of EN 301 545-2";

TR 101 545-5: "Guidelines for the Implementation and Use of TS 101 545-3".

Introduction

The present document gives guidelines for the implementation of the second generation of Digital Video Broadcasting (DVB) interaction channel for Satellite Distribution System (also known as DVB-RCS2: Return Channel via Satellite), particularly related to lower layer specifications [i.1]. It also describes the specification for geostationary satellite interactive system [i.3].

The present document provides some examples of implementation details related either with the physical (e.g. code performance, link layer encapsulation, link budget) or the medium access control (e.g. use of capacity request categories) layers. It draws attention to the technical questions that need to be answered in the implementation of the DVB-RCS2 specification and setting up a DVB-RCS2 network and offers some guidance in finding answers to them.

Outline of the present document

Clause 4 offers reference models for different scenarios of DVB-RCS2 use.

Clause 5 provides implementation guidelines for the Forward Link and the lower layer signalling.

Clause 6 offers implementation guidelines for the Return Link.

Clause 7 describes guidelines for implementation the M&C functions supported by the lower layer signalling.

Clause 8 offers guidelines for TRANSEC implementation for different application profiles.

Clause 9 offers guidelines for RCST deployment such as information concerning the most relevant international regulations and recommendations (ITU, ETSI, DVB, etc.).

Clause 10 offers guidelines for system deployment, performance results for physical layer, efficiency figures of the return link encapsulation, link budget examples as well as system capacity evaluation examples.

Annex A is a specification of an alternative design for implementation of even more flexible CPM waveform.

Annex B provides some examples of CRDSA implementation and performance.

Annex C describes a possible standards evolution introducing SC-FDMA.

Annex D describes a possible enhancement where timeslots are shared by specific transmitters in transparent mesh systems.

Annex E describes alternative methods for energy spreading of the forward link.

Annex F describes the legacy method for return link energy spreading using burst repetition.

Annex G specifies the ODU control protocol inherited from the guidelines document for the first generation of the DVB-RCS Standard.

1 Scope

The present document provides implementation guidelines for equipment and systems intended to comply with [i.1]. It also provides designs that may be used to supplement the normative specifications provided in [i.1]. Such designs could evolve into being a part of the normative specifications in the future.

2 References

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

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

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

2.1 Normative references

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

Not applicable.

2.2 Informative references

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

[i.1] ETSI EN 301 545-2 (V1.1.1): "Digital Video Broadcasting (DVB); Second Generation DVB Interactive Satellite System (DVB-RCS2); Part 2: Lower Layers for Satellite standard".

[i.2] ETSI EN 302 307: "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2)".

[i.3] ETSI TS 101 545-1: "Digital Video Broadcasting (DVB); Second Generation DVB Interactive Satellite System (DVB-RCS2); Part 1: Overview and System Level specification".

[i.4] ETSI TS 101 545-3: "Digital Video Broadcasting (DVB); Second Generation DVB Interactive Satellite System (DVB-RCS2); Part 3: Higher Layers Satellite Specification".

[i.5] Guidelines for 64-bit global identifier (EUI-64) Registration Authority.

NOTE: Available at http://standards.ieee.org/regauth/oui/tutorials/EUI64.html.

[i.6] ETSI EN 301 459: "Satellite Earth Stations and Systems (SES); Harmonized EN for Satellite Interactive Terminals (SIT) and Satellite User Terminals (SUT) transmitting towards satellites in geostationary orbit in the 29,5 GHz to 30,0 GHz frequency bands covering essential requirements under article 3.2 of the R&TTE Directive".

[i.7] ETSI EN 301 428: "Satellite Earth Stations and Systems (SES); Harmonized EN for Very Small Aperture Terminal (VSAT); Transmit-only, transmit/receive or receive-only satellite earth stations operating in the 11/12/14 GHz frequency bands covering essential requirements under article 3.2 of the R&TTE directive".

[i.8] ETSI EN 301 427: "Satellite Earth Stations and Systems (SES); Harmonized EN for Low data rate Mobile satellite Earth Stations (MESs) except aeronautical mobile satellite earth stations, operating in the 11/12/14 GHz frequency bands covering essential requirements under article 3.2 of the R&TTE directive".

[i.9] ETSI EN 302 186: "Satellite Earth Stations and Systems (SES); Harmonized EN for satellite mobile Aircraft Earth Stations (AESs) operating in the 11/12/14 GHz frequency bands covering essential requirements under article 3.2 of the R&TTE Directive".

[i.10] ETSI EN 302 340: "Satellite Earth Stations and Systems (SES); Harmonized EN for satellite Earth Stations on board Vessels (ESVs) operating in the 11/12/14 GHz frequency bands allocated to the Fixed Satellite Service (FSS) covering essential requirements under article 3.2 of the R&TTE directive".

[i.11] ETSI EN 302 448: "Satellite Earth Stations and Systems (SES); Harmonized EN for tracking Earth Stations on Trains (ESTs) operating in the 14/12 GHz frequency bands covering essential requirements under article 3.2 of the R&TTE directive".

[i.12] ETSI EN 302 977: "Satellite Earth Stations and Systems (SES); Harmonized EN for Vehicle-Mounted Earth Stations (VMES) operating in the 14/12 GHz frequency bands covering the essential requirements of article 3.2 of the R&TTE directive".

[i.13] FCC Part 25-Satellite Communications, § 25.221: "Blanket Licensing provisions for Earth Stations on Vessels (ESVs) receiving in the 3700-4200 MHz (space-to-Earth) frequency band and transmitting in the 5925-6425 MHz (Earth-to-space) frequency band, operating with Geostationary Satellites in the Fixed-Satellite Service. § 25.222 Blanket Licensing provisions for Earth Stations on Vessels (ESVs) receiving in the 10.95-11.2 GHz (space-to-Earth), 11.45-11.7 GHz (space-to-Earth), 11.7-12.2 GHz (space-to-Earth) frequency bands and transmitting in the 14.0-14.5 GHz (Earth-to-space) frequency band, operating with Geostationary Satellites in the Fixed-Satellite Service".

[i.14] Recommendation ITU-R M.1643: "Technical and operational requirements for aircraft earth stations of aeronautical mobile-satellite service including those using fixed-satellite service network transponders in the band 14-14.5 GHz (Earth-to-space)".

[i.15] ETSI EN 301 358: "Satellite Earth Stations and Systems (SES); Satellite User Terminals (SUT) using satellites in geostationary orbit operating in the 19,7 GHz to 20,2 GHz (space-to-earth) and 29,5 GHz to 30 GHz (earth-to-space) frequency bands".

[i.16] ETSI EN 301 489-12: "Electromagnetic compatibility and Radio spectrum Matters (ERM); ElectroMagnetic Compatibility (EMC) standard for radio equipment and services; Part 12: Specific conditions for Very Small Aperture Terminal, Satellite Interactive Earth Stations operated in the frequency ranges between 4 GHz and 30 GHz in the Fixed Satellite Service (FSS)".

[i.17] ETSI ETS 300 784: "Satellite Earth Stations and Systems (SES); Television Receive-Only (TVRO) satellite earth stations operating in the 11/12 GHz frequency bands".

[i.18] CENELEC EN 61319-1: "Interconnections of satellite receiving equipment Part 1: Europe".

[i.19] CENELEC EN 50083 series: "Cable networks for television signals, sound signals and interactive services".

[i.20] DiSEqC Bus Specification, Version 4.2, EUTELSAT: "DiSEqC Bus Specification".

[i.21] NIST Special Publication 800-38A, 2001 edition: "Recommendation for Block Cipher Modes of Operation - Methods and Techniques".

[i.22] IETF RFC 4945: "The Internet IP Security PKI Profile of IKEv1/ISAKMP, IKEv2, and PKIX".

[i.23] IETF RFC 4880: "OpenPGP Message Format".

[i.24] Recommendation ITU-T X.509: "Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks", August 2005 and later corrigenda".

[i.25] Rivest, R.; A. Shamir; L. Adleman: "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", Communications of the ACM 21 (2): 120-126.

[i.26] IEEE 802.3: "Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications".

[i.27] IETF RFC 5911: "New ASN.1 Modules for Cryptographic Message Syntax (CMS) and S/MIME".

[i.28] ETSI TS 101 162: "Digital Video Broadcasting (DVB); Allocation of identifiers and codes for Digital Video Broadcasting (DVB) systems".

[i.29] George Marsaglia, "Xorshift RNGs", Journal of Statistical Software (Vol.8 Issue 14), July 2003.

[i.30] IETF RFC 3629: "UTF-8, a transformation format of ISO 10646", November 2003.

[i.31] RSA Laboratories, "PKCS #5 v2.1: Password-Based Cryptography Standard", October 5, 2006.

NOTE: Available from http://www.rsa.com/rsalabs/node.asp?id=2127.

[i.32] National Institute of Standards and Technology, "Secure Hash Standard", FIPS Pub. 180-2, August 1, 2002.

[i.33] IETF RFC 4648: "The Base16, Base32, and Base64 Data Encodings".

[i.34] Federal Information Processing Standards Publication 140-2, "Security Requirements for Cryptographic Modules", May 25, 2001 and later amendments.

[i.35] Recommendation ITU-R P.1623-1: "Prediction method of fade dynamics on Earth-space paths", 2005-03r. 1988.

[i.36] ECC/DEC(06)02, 2006: "Electronic Communications Committee, ECC Decision of 24 March 2006 on Exemption from Individual Licensing of low e.i.r.p. satellite terminals (LEST) operating within the frequency bands 10.70 - 12.75 GHz or 19.70 - 20.20 GHz Space-to-Earth and 14.00 - 14.25 GHz or 29.50 - 30.00 GHz Earth-to-Space".

NOTE: Available at http://www.erodocdb.dk/docs/doc98/official/pdf/ECCDec0602.pdf.

[i.37] ECC/DEC(06)03, 2006: " Electronic Communications Committee, ECC Decision of 24 March 2006 on Exemption from Individual Licensing of High e.i.r.p. satellite terminals (HEST) operating within the frequency bands 10.70 - 12.75 GHz or 19.70 - 20.20 GHz Space-to-Earth and 14.00 - 14.25 GHz or 29.50 - 30.00 GHz Earth-to-Space".

NOTE: Available at http://www.erodocdb.dk/docs/doc98/official/pdf/ECCDec0603.pdf.

[i.38] Rapp: "Effects of the HPA-nonlinearity on a 4-DPSK/OFDM signal for a digital sound broadcasting system" in: Second European Conf. on Sat. Comm., 22. - 24.10.91, Liege, Belgium., 1991.

[i.39] B. E. Rimoldi: "A decomposition approach to CPM", IEEE Trans. Inform. Theory, vol. 34, pp. 260-270, March 1988.

[i.40] A. Barbieri and G. Colavolpe: "Simplified soft-output detection of CPM signals over coherent and phase noise channels," IEEE Trans. Wireless Commun., vol. 6, pp. 2486-2496, July 2007.

[i.41] L. Bahl, J. Cocke, F. Jelinek and J. Raviv: "Optimal decoding of linear codes for minimizing symbol error rate," IEEE Trans. Info. Theory, vol. IT-20, no. 2, pp. 284-287, March 1974.

[i.42] D.C. Rife and R.R. Boorstyn: "Single tone parameter estimation from discrete-time observations", IEEE Trans. Inform. Theory, vol. 20, pp. 591-598, September 1974.

[i.43] Directive 1999/5/EC of the European Parliament and of the Council of 9 March 1999 on radio equipment and telecommunications terminal equipment and the mutual recognition of their conformity (R&TTE Directive).

[i.44] Ichikawa M., Hara T., Saito M., Okada M. and Yamamoto H. "Evaluation of BER degradation due to phase nonlinearity of SSPA on MC-CDMA signals", Personal Wireless Communications, 2005. ICPWC 2005. Page(s): 533 - 536.

[i.45] M. Angelone, A. Ginesi, E. Re and S. Cioni: "Performance of a Combined Dynamic Rate Adaptation and Adaptive Coding Modulation Technique for a DVB-RCS2 System", Proceedings of ASMS/SPSC Conference 2012, Baiona, Spain, Pages 124-131.

NOTE: Available at http://ieeexplore.ieee.org.

[i.46] ETSI EN 301 790: "Digital Video Broadcasting (DVB); Interaction channel for satellite distribution systems".

[i.47] ETSI TR 101 790 (V1.4.1): "Digital Video Broadcasting (DVB); Interaction channel for Satellite Distribution Systems; Guidelines for the use of EN 301 790".

[i.48] M. Holzbock, A. Jahn, O. Gremillet, E. Lutz: "Aeronautical channel characterization measurements at K Band," in Proceedings 4th Ka Band Utilization Conference", Venice, Italy, pp. 263-269, November 1998.

[i.49] S. Scalise, H. Ernst and G. Harles: "Measurement and Modeling of the Land Mobile Satellite Channel at Ku-Band", IEEE Transaction on Vehicular Technology, Vol. 57, No. 2 March 2008.

[i.50] E. Kubista, F. Perez Fontan, M. A. Vazquez Castro, S. Buonomo, B. R. Arbesser-Rastburg, J.P.V. Poiares Baptista: "Ka-Band Propagation Measurements and Statistics for Land Mobile Satellite Applications", IEEE Transaction on Vehicular Technology, Vol. 49, No. 3 May 2000.

[i.51] F. Perez Fontan, M. Vazquez Castro, C. Enjamio Cabado, J. Pita Garcia, and E. Kubista, "Statistical modelling of the LMS channel," IEEE Transactions on Vehicular Technology, vol. 50, pp. 1549-1567, November 2001.

[i.52] P.A. Laurent: "Exact and approximate construction of digital phase modulations by superposition of amplitude modulated pulses (AMP)", IEEE Trans. Commun., vol. 34, pp. 150-160, February 1986.

[i.53] A. Barbieriand G. Colavolpe: "Simplified Soft-Output Detection of CPM Signals Over Coherent and Phase Noise Channels", IEEE Trans. on Wireless Comm., vol. 6, n. 7, July 2007.

[i.54] G. Colavolpe and R. Raheli: "Reduced-complexity detection and phase synchronization of CPM signals," IEEE Trans. Commun., vol. 45, pp. 1070-1079, September 1997.

[i.55] N. Abramson: "The Throughput of Packet Broadcasting Channels", IEEE Trans. Communications, vol. COM-25, no. 1, pp. 117-128, January 1977.

[i.56] G. L. Choudhury and S. S. Rappaport: "Diversity ALOHA - A Random Access Scheme for Satellite Communications,"IEEE Trans. on Comm., vol. COM-31, March 1983, pp. 450-457.

[i.57] E. Casini, R. De Gaudenzi and O. del Rio Herrero: "Contention Resolution Diversity Slotted Aloha (CRDSA): an Enhanced Random Access Scheme for Satellite Access Packet Networks", IEEE Transactions on Wireless Communications, vol. 6, no. 4, pp. 1408-1419, April 2007.

[i.58] Oscar del Rio Herrero and Riccardo De Gaudenzi: "A High-Performance MAC Protocol for Consumer Broadband Satellite Systems", In the Proc. of 27th AIAA International Communications Satellite Systems Conference, June 1-4 2009, Edinburgh (United Kingdom).

[i.59] R. De Gaudenzi and O. Del Rio-Herrero: "Advances in Random Access protocols for satellite networks", International Workshop on Satellite and Space Communications, 2009, IWSSC 2009, Siena, Italy September 9-11, 2009, pp. 331-336.

[i.60] "ESA Project on Advanced Modem Prototype for Interactive Satellite Terminals".

NOTE: Available at: http://telecom.esa.int/telecom/www/object/index.cfm?fobjectid=30949.

[i.61] Del Rio Herrero and R. De Gaudenzi, US Patent 7,990,874: "Methods, Apparatuses and System for Asynchronous Spread Spectrum Communication", August 2, 2011.

[i.62] G. Liva: "Graph-Based Analysis and Optimization of Contention Resolution Diversity Slotted ALOHA", IEEE Trans. On Comm., Vol. 59, Issue 2, February 2011, pp. 477-487.

[i.63] Final Report: "Motorised antenna mount for bi-directional satellite terminals", ESA Study Contract No. 21632/08/NL/AD.

[i.64] ETSI TS 136 211: "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation (3GPP TS 36.211 version 8.9.0 Release 8)".

[i.65] Digital Video Broadcasting (DVB) - Next Generation Handheld (NGH); Frame structure channel coding and modulation.

[i.66] H. G. Myung, J. Lim and D. J. Goodman: "Single carrier FDMA for uplink wireless transmission", IEEE Vehicular Tech. Magazine, pp. 30-38, June, 2009.

[i.67] M. Morelli and U. Mengali: "An improved frequency offset estimator for OFDM applications," IEEE Commun. Letters, Vol. 3, No. 3, Mar. 1999, pp. 75-77.

[i.68] Giambene Giovanni (Ed.,): "Resource Management in Satellite Networks Optimization and Cross-Layer Design", 2007, Springer.

[i.69] Recommendation ITU-T G.1030 (2005): "Estimating end-to-end performance in IP networks for data applications".

[i.70] ANSI/IEEE Standard 754 (1985): "IEEE Standard for Binary Floating-Point Arithmetic".

[i.71] IETF RFC 4326: "Unidirectional Lightweight Encapsulation (ULE) for Transmission of IP Datagrams over an MPEG-2 Transport Stream (TS)".

[i.72] IETF RFC 5163: "Extension Formats for Unidirectional Lightweight Encapsulation (ULE) and the Generic Stream Encapsulation (GSE)".

[i.73] Recommendation ITU-R S.728.1: "Maximum permissible level of off-axis e.i.r.p. density from very small aperture terminals (VSATs)".

[i.74] EN 303 978: "Satellite Earth Stations and Systems (SES); Harmonized EN for Earth Stations on Mobile Platforms (ESOMP) transmitting towards satellites in geostationary orbit in the 27,5 GHz to 30,0 GHz frequency bands covering the essential requirements of article 3.2 of the R&TTE Directive".

[i.75] ETSI TS 102 602: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; Connection Control Protocol (C2P) for DVB-RCS; Specifications".

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in EN 301 545-2 [i.1] apply.

3.2 Symbols

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

α	Roll-off factor
A, B	Input sequences to the turbo encoder
C_1	Circulation state of the turbo encoder in the natural order
C_2	Circulation state of the turbo encoder in the interleaved order
E_b/N_0	Ratio between the energy per information bit and single sided noise power spectral density
E_s/N_0	Ratio between the energy per transmitted symbol and single sided noise power spectral density
f_0	Carrier frequency
f_N	Nyquist frequency
H(f)	Raised Cosine filters frequency transfer function
I, Q	In-phase, Quadrature phase components of the modulated signal
K/N	GSPC code rate

N_b	GSPC sub-blocks number
$N_{R,max}$	Number of replicas in a frame
Nrand	12-bit random number used as a random seed value during CRDSA frame decoding
N_{slots}	Number of the slots in the frame
$p_1, p_2, \ldots, p_{NR,max}$	Vector that contains the $N_{R,max}$ indices of the slots containing the burst replicas
$p_{d_j-1}, \cdots, p_0$	GSPC code parity bits
R, k/n	Burst code rate
R_s	Symbol rate corresponding to the bilateral Nyquist bandwidth of the modulated signal
S	State of the turbo encoder
S_x	Symbol
T_s	Symbol period
u_x	Bits
X	GSPC code information word
X(D)	GSPC code information polynomial
$x_{K-1}, \cdots, x_0$	GSPC code information bits
Z_1	Output sequence of the puncturing for the encoder in the natural order
Z_2	Output sequence of the puncturing for the encoder in the interleaved order

3.3 Abbreviations

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

16QAM	16-ary QAM
8PSK	8-ary PSK
AC	Allocation Channel
ACC	Acquisition Ciphertext Channel
ACI	Adjacent Channel Interference
ACK	ACKnowledgement
ACM	Adaptive Coding and Modulation
AES	Advanced Encryption Standard
AF	Assured Forwarding
ALPDU	Addressed Link Protocol Data Unit
AM/AM	Amplitude Modulated to Amplitude Modulated characteristic (Power Transfer)
AM/PM	Amplitude Modulation/ Phase Modulation (Phase Transfer)
AMSS	Aeronautical Mobile Satellite Service
ANT	Antenna (Subsystem)
APSK	Amplitude and Phase Shift Keying (modulation)
ASCII	American Standard Code for Information Interchange
ASIC	Application Specific Integrated Circuit
ATM	Asynchronous Transfer Mode
AVBDC	Absolute VBDC
AWGN	Additive White Gaussian Noise
BA	Behaviour Aggregate
BB	Base-Band
BBFRAME	BaseBand FRAME
BCH	Bose - Chaudhuri - Hocquenghem (code)
BCJR	Bahl, Cocke, Jelinek and Raviv (algorithm)
BCT	RL Broadcast Configuration Table
BE	Best Effort
BPSK	Binary PSK
bslbf	bit string, left bit first
BTP	Burst Time Plan
BTU	Bandwidth-Time Unit
BUC	Block Up Converter
CA	Connectivity Aggregate
CA/KG	Certificate Authority/ Key Generator
CAZAC	Constant Amplitude Zero Auto Correlation
CBC	Cipher Block Chaining

CC	Convolutional Coding
CC-CPM	Convolutional Code – Continuous Phase Modulation
CCITT	(The) International Telegraph and Telephone Consultative Committee
CCM	Constant Coding and Modulation
CEPT	European Conference of Postal and Telecommunications Administrations
CFB	Cipher Feedback
CMF	Control and Monitoring Functions
CMOS	Charge-coupled Metal Oxide Silicon
CNR	Carrier to Noise power ratio
CP	Cyclic Prefix
CPE	Continuous Phase Encoder
CPM	Continuous Phase Modulation (or Modulator)
CR	Capacity Request
CRA	Constant Rate Assignment
CRC	Cyclic Redundancy Check
CRDSA	Contention Resolution DSA
CW	Continuous Wave
DA	Dedicated Access
DA-AC	Dedicated Access Allocation Channel
DAAF	Digital Anti-Aliasing Filter
DAMA	Demand Assigned Multiple Access
DC	Direct Current
DCC	Dynamic Ciphertext Channel
DCP	Dynamic Connectivity Protocol
DFL	Data Field Length
DFT	Discrete Fourier Transform
DNS	Domain Name Server
DRA	Data Rate Adaptation
DSA	Diversity Slotted Aloha
DSCP	Differentiated Service Code Point
DSSS	Direct Sequence Spectrum Spreading
DVB	Digital Video Broadcasting
DVB-RCS2	Digital Video Broadcast Return Channel via Satellite 2nd generation
ECB	Electronic Code Book
ECC	Electronic Communication Committee (of the CEPT)
EF	Expedited Forwarding
EIRP	Effective Isotropic Radiated Power
EMC	ElectroMagnetic Compatibility
ESV	Earth Stations (on board) Vessels
FCA	Free Capacity Assignment
FCC	Federal Communication Commission
FCT2	Frame Configuration Table 2
FEC	Forward Error Correction
FER	Frame Error Ratio
FFT	Fast Fourier Transform
FL	Forward Link
FLS	Forward Link Signalling
FPDU	Frame PDU
FPGA	Field-Programmable Gate Array
FRSS	Frame Repetition Spectrum Spreading
FS	Fixed Service
FSS	Fixed Satellite Service
GPS	Global Positioning Systems
GS	Generic Stream
GSE	Generic Stream Encapsulation
GSO	GeoStationary (earth) Orbit
GSPC	Generic Sub-block Polynomial Code
GW	Gateway
HL	Higher Layer
HLC	Higher Layer Classification
HLS	Higher Layer Specifications
HSM	Hub Security Module

HW	HardWare
IANA	Internet Assigned Numbers Agency
IB	Installation Burst
IBO	Input Back Off
ICT	Interleaver Configuration Table
ID	Identifier
IDU	Indoor Unit
IERS	International Earth Rotation and Reference System Service
IF	Intermediate Frequency
IFDMA	Interleaved Frequency Division Multiple Access
IFFT	Inverse Fast Fourier Transform
IFL	Inter-facility Link
ISI	Input Stream Identifier
ISSYI	Input Stream SYnchronizer Indicator
IV	Initialization Vectors
KEK	Key Encrypting Keys
L2S	Lower Layer Signalling
LAN	Local Area Network
LDPC	Low Density Parity Check
LFDMA	Localized Frequency Division Multiple Access
LL	Link Layer
LL	Lower Layer
LLC	Lower Layer Classification
LL-FEC	Link Layer Forward Error Correction
LLS	(DVB-RCS2) Lower Layer Specification
LM	Linear Modulation (or Modulator)
LMSS	Land Mobile Satellite Service
LNB	Low Noise Block
LO	Local Oscillator
LOS	Line Of Sight
LPF	Low Pass Filtering
LSB	Least Significant Bit
LUT	Look-up Table
MAC24	A 24 bit MAC address
MAP	Maximum A Posteriori
MATYPE	Mode Adaptation TYPE
MC	Mesh Controller
MCD	Multi Carrier Demodulator
MECH	Mechanical Subsystem
MES	Mobile Satellite Earth Stations
MF-TDMA	Multi-Frequency TDMA
MIB	Management Information Base
MM	Memoryless Modulator
MMSS	Maritime Mobile Satellite Service
MODCOD	Modulation and Coding
MPEG	Moving Pictures Expert Group
MSB	Most Significant Bit
MSS	Mobile Satellite Service
MTU	Maximum Transmission Unit
NACK	Negative Acknowledgment
NCC	Network Control Centre
NCC/GW	Network Control Centre/Gateway
NCR	Network Clock Reference
NIT	Network Information Table
NIU	Network Interface Unit
NLOS	Non-Line-Of-Sight mobile
NMC	Network Management Centre
NMS	Network Management System
NOC	Network Operator Centre
NPD	Null Packet Deletion
NRM	Network Resource Model
NSC	Network Security Controller

NTP	Network Time Protocol
OBO	Output Back-Off
OBP	On Board Processing
ODU	OutDoor Unit
OFDMA	Orthogonal Frequency Division Multiple Access
PAM	Pulse Amplitude Modulation
PAPR	Peak to Average Power Ratio
PCI	Peripheral Component Interconnect
PCR	(MPEG-2) Program Clock Reference
PDU	Protocol Data Unit
PER	Packet Error Ratio
PFD	Power Flux Density
PHB	Per Hop Behavior
PHY	Physical (Layer)
PL	Physical Layer
PLL	Phase Lock Loop
PLR	Packet Loss Ratio
PPDU	Payload-adapted PDU
PRG	Pseudo-Random Generator
PSD	Power Spectral Density
PSI	Program Specific Information
PSK	Phase Shift Keying
PSU	Power Supply Unit
PWD	Password
PWK	Pulse Width Keying
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RA	Random Access
RA-AC	Random Access Allocation Channel
RA-SIG	Random Access Signalling
RBDC	Rate Based Dynamic Capacity
RC	Raised-Cosine
RCS	Return Channel over Satellite
RCST	RCS Terminal (compliant with DVB-RCS2)
REC	Rectangular
RF	Radio Frequency
RFC	Request For Comments
RL	Return Link
RLE	Return Link Encapsulation
RMS	Root Mean Square
RMT	RCS Map Table
RNG	Random Number Generator
RO	Roll-Off
RRM	Radio Resource Management
RSC	Recursive Systematic Convolutional (code)
RSM	RCST Security Module
RSMS	Regenerative Satellite Multimedia System
RT	Real Time
RX	Receiver
SA	Slotted Aloha
SAT	Satellite
SCADA	Supervisory Control and Data Acquisition
SCT	Superframe Composition Table
SDU	Service Data Unit
SF	Spreading Factor (in Annex E), or Super-Frame (in Clause 6)
SFS	Superframe Sequence
SGW	Satellite Gateway
SHA	Secure Hash Algorithm
SI	Service Information
SISO	Single Input Single Output
SLA	Service Level Agreement

SNIR	Signal to Noise (plus) Interference Ratio
SNR	Signal to Noise Ratio
SOF	Start of Frame
spfmsbf	single precision floating-point, most significant bit first
SPT	Satellite Position Table
SRM	Satellite Remote MODEM
SRS	Space Research Service
SSB	Single Side Band
ST	Satellite Terminal
SVN	Satellite Virtual Network
SW	SoftWare
SYNC	SYNChronization
SYNCD	SYNC Distance
TBTP2	Terminal Burst Time Plan 2
TC	Turbo Coding
TC-LM	Turbo-Coded Linear Modulation (scheme in DVB-RCS2)
TCP	Transport Control Protocol
TD	Total Distortion
TDM	Time Division Multiplex
TDMA	Time Division Multiple Access
TIM-B	Terminal Information Message Broadcast
TIM-U	Terminal Information Message Unicast
TOD	Time of Day
TRANSEC	TRANSmission SECurity
TRF	Traffic
TS	Transport Stream
TS/GS	Transport Stream/Generic Stream
TSS	Time Slot Sharing
TWT	Travelling Wave Tube
TWTA	Travelling Wave Tube Amplifier
TX	Transmitter
uimsbf	unsigned integer most significant bit first
UIU	User Interface Unit
UPL	User Packet Length
UW	Unique Word
VACP^{TM}	VSAT Antenna Control Protocol
VBDC	Volume Based Dynamic Capacity
VCM	Variable Coding and Modulation
VMES	Vehicle-Mounted Earth Stations
XPD	Cross Polarization Discrimination

4 Reference Model

An overview of DVB-RCS2 system scenarios and network topologies is described in [i.3]. Reference models to realize such satellite networks could include different interconnections among Network Control Centre, Traffic Gateway(s), Feeder(s) and Terminals.

In practice not all these interconnections will be implemented. Also, some functional blocks may be co-located. This clause describes therefore the network architectures that are more likely to be implemented for the service provision.

4.1 Architecture with co-located NCC, Gateway and Feeder

The simplest architecture is an interactive satellite network with a single Traffic Gateway and a single Feeder co-located in an Earth Station (see Figure 4.1). The Network Control Centre is possibly also collocated.

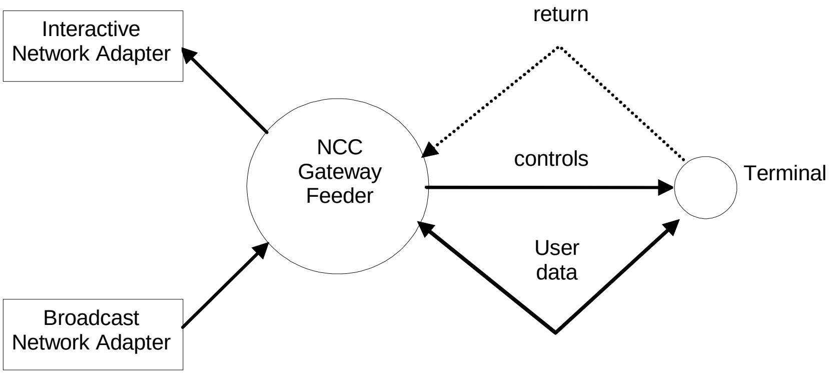


Figure 4.1: Architecture with a single gateway and feeder (collocated)

This Earth Station has both an Interactive Network Adapter and a Broadcast Network Adapter. It generates the forward link signal, including user data and the control and timing signals needed for the operation of the Satellite Interactive Network. It receives the RCST return signals, provides interactive services and/or connections to external service providers and networks and it provides monitoring, accounting and billing functions.

4.2 Architecture with multiple feeders

When more Feeders exist in the interactive satellite network, the terminals should be able to switch from one to another, without losing network synchronization (see Figure 4.2). In order to achieve this, the following network architecture is envisaged. Terminals are equipped with at least two receivers. One receiver is continuously tuned to the forward link from a "primary" Feeder, the one which includes the control and timing signals and which provides monitoring, accounting and billing. The other receiver(s) can be tuned to different signals transmitted by "secondary" feeds to receive user data. The capability of the ODU to receive separate signals is the only limitation.

In this configuration, terminals tuned to different "primary" Feeders (most likely belonging to different networks), might receive information from the same "secondary" Feeder(s).

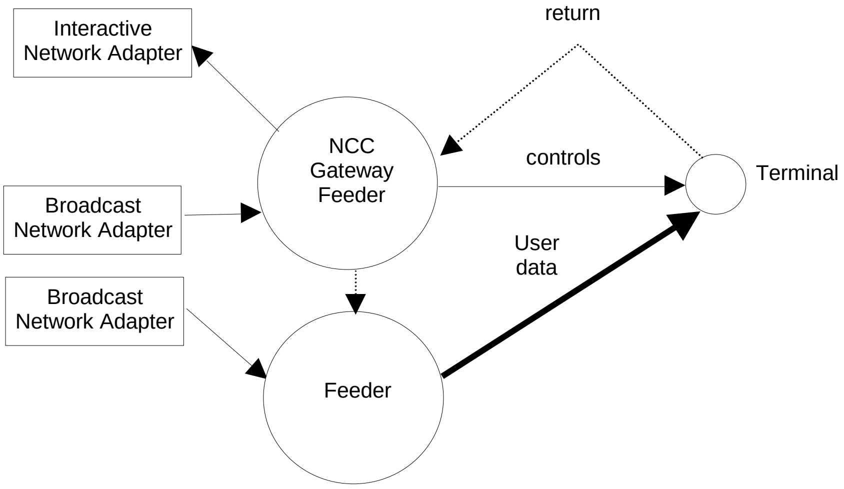

Interactive Network Adapter
return
NCC Gateway Feeder
controls
Terminal
Broadcast Network Adapter
User data
Broadcast Network Adapter
Feeder

Figure 4.2: Architecture with more than one feeder

4.3 Architecture with transparent mesh connectivity

By incorporating one or more TDMA burst receivers the RCST will be capable of receiving TDMA bursts as well as transmitting them. This allows RCSTs to communicate directly over a bent-pipe satellite, as indicated in Figure 4.3, as well as simultaneously operating according to the architectures of Figures 4.1 and 4.2. Figure 4.3 shows the architecture of Figure 4.1 extended with transparent mesh capability.

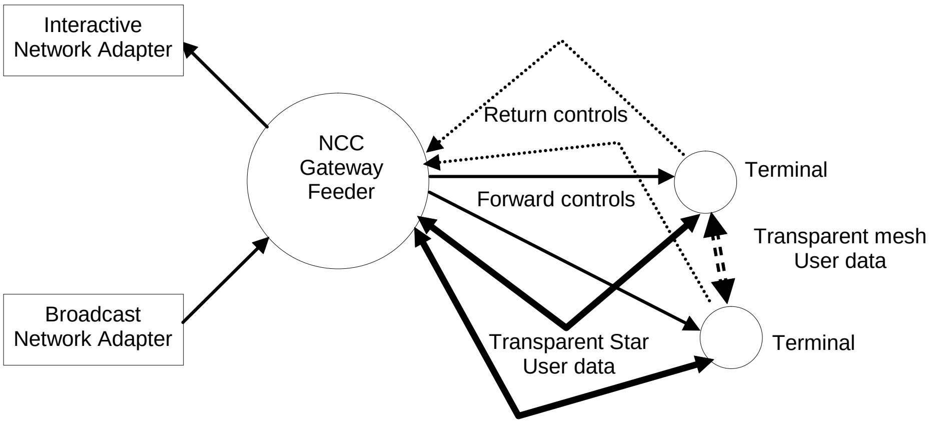

Interactive Network Adapter
Broadcast Network Adapter
NCC Gateway Feeder
Return controls
Forward controls
Terminal
Transparent mesh User data
Terminal
Transparent Star User data

Figure 4.3: Architecture with transparent mesh connectivity

4.4 Architecture with regenerative satellites

Figure 4.4 highlights a reference architecture for a regenerative satellite network where on-board processing capability of the satellite allows for an efficient peer-to-peer connectivity of RCSTs.

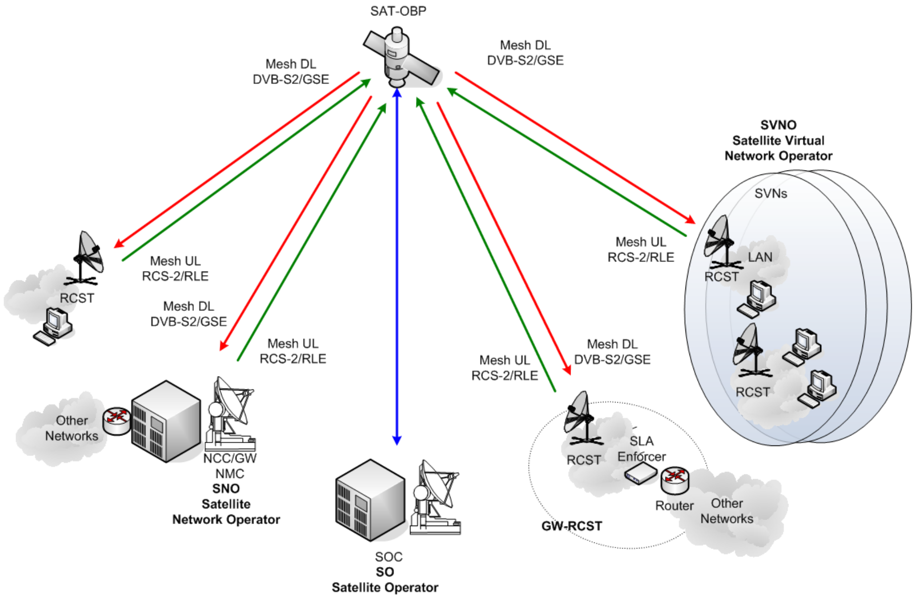


Figure 4.4: Regenerative Satellite Network Reference Model

The Onboard Processors are classified as:

- **Regenerative with onboard switching** that can provide full traffic re-arrangement for point-to-point connections between terminals in a mesh network. The onboard processor can also be configured to support point-to-multipoint, multi-point-to-point connections and/or concentration /multicasting /broadcasting through flexible routing/switching between input and output ports.

- **Regenerative without onboard switching** which is particularly attractive when the number of uplink and/or downlink beams is relatively small and the requirements for onboard traffic arrangement are moderate. In such cases the requirements for concentration and/or multicasting/ multiplexing type of connectivity prevail.

- **Regenerative in conjunction with transparent repeater** which is based on a hybrid payload including both transparent and regenerative onboard switching repeaters. The terminals are connected to the RSMS network through the transparent repeater. Point-to-point connectivity between terminals is provided by the OBP Processor, hereafter called *"mesh processor"*.

The functional requirements of the OBP processor are:

- Receive all traffic and control data sent by the terminals.
- Receive all traffic and control data sent by the NCC.
- Extract the traffic data to be sent on the downlink within DVB- S2 format and route them to the appropriate output(s) towards the receiving terminals.
- Generate/extract the control data to be sent to the NCC and route them to the appropriate output(s).
- Format downlink streams including all the necessary downlink signalling messages in DVB-RCS2 compatible DVB-S2 format and route/switch them to the appropriate output(s).

Different OBP implementations result from the apportionment of the MAC functions between onboard processor and on-ground entities; the RCS terminals and NCC. These are described in Table 4.1.

Table 4.1: MAC Functions Partitioning in case of regenerative OBP satellite systems

MAC Function	Network Entity	Comments
Data encapsulation / de-capsulation	**OBP**	See OBP switching / routing modes in clause 4.4.1.
Routing_Label Extraction	**OBP**	See OBP switching / routing modes in clause 4.4.1.
Frame Format	**OBP**	See OBP switching / routing modes in clause 4.4.1.
Synchronization and power control	**OBP/NCC**	***On board measurements***
NCR generation and insertion	**OBP/NCC**	
Resource control and management	**RCS (Capacity requests)/NCC/OBP**	Performed on ground in case of Hybrid regenerative P/L with "mesh processor". Performed also on board in case Traffic manager is implemented on board
RCS configuration and management	**NCC**	On ground
Logon	**NCC/OBP**	On board measurements
Network Configuration	**Network Configuration**	On ground

4.4.1 On board switching requirements

A regenerative system requires onboard routing or switching of signals between input and output ports. Different on board switching architectures can be used; circuit-switched, frame-switched, packet or cell switched architecture.

The On-Board Processor (OBP) in a regenerative system uses a DVB-RCS2 air interface on the uplink (return link) and a DVB-S2 in the downlink (forward link), providing the modulation, coding and framing functions. Different architectures are possible on the additional functions (multiplexing, table-switching, routing, etc.) or the QoS support.

In case of regenerative payload performing packet or cell switching, there are two types of information identified as necessary to perform routing/switching:

- ***Addressing information*** to identify the destination RCST; and
- ***Control (routing) information*** to the Onboard Processor (OBP) to perform routing/switching.

The increasing need of addressing and controlling/routing information is attributable to specific system design assumptions:

- The multi-beam coverage and the multi-port onboard switching matrix increase the number of possible routes.
- The support of multiple levels of Quality of Service.
- The multi-operator context: the inter-operability is increasing the routing possibilities and limits the reuse of the identifiers.

- The increasing number of simultaneous connections managed by a terminal (the number of customer premises equipment connected to this terminal).

The mesh traffic between terminals is only handled by the terminals but controlled, monitored and allocated by the NCC. There are cases of regenerative satellite systems where the traffic manager functions are split between the NCC and the on-board processor. In this latter case, the on board traffic manager plays an active role for the control, monitoring and allocation of resources to terminals.

A regenerative satellite network can be configured to support both star and mesh topologies based on DVB-RCS2 specification for the return channel and DVB-S2 specification for the forward channel.

As for a transparent satellite network, the regenerative satellite network in star topology supports access traffic to/from a gateway. In a regenerative architecture this star topology features some enhanced access network flexibility, such as interconnecting terminals to multiple gateways and/or multiplexing the star traffic with mesh traffic on a same given downlink carrier for the destination beam.

The regenerative satellite network is characterized by the capability of star and mesh single-hop communications, and are composed of one or more beams. For multi-beam regenerative systems, cross-connectivity can be supported.

Following RLE/GSE RCS2 encapsulation, different OBP switching modes can be observed in a regenerative satellite system:

- Carrier/timeslot switching: used mainly for NCC signalling and prosumer traffic, with no multiplexing gain
- Burst switching:
    - Based on RLE in uplink and downlink: no decapsulation/re-assembly needed
    - All RLE packets in one burst should share destination (and ACM/QoS)
- Packet switching:
    - Burst label not needed
    - No fragmentation of packets is considered (high impact in OBP)
    - Packet length should be chosen to fit in burst length
- Fragment label switching: Fragment label is used for switching and sender identification, with two profiles:
    - UL-RLE/DL-RLE the most efficient and simple
    - UL-RLE/DL-GSE requires more complexity at OBP level
- Layer 3 switching:
    - OBP capable of fully de-encapsulation - encapsulation up to IP layer

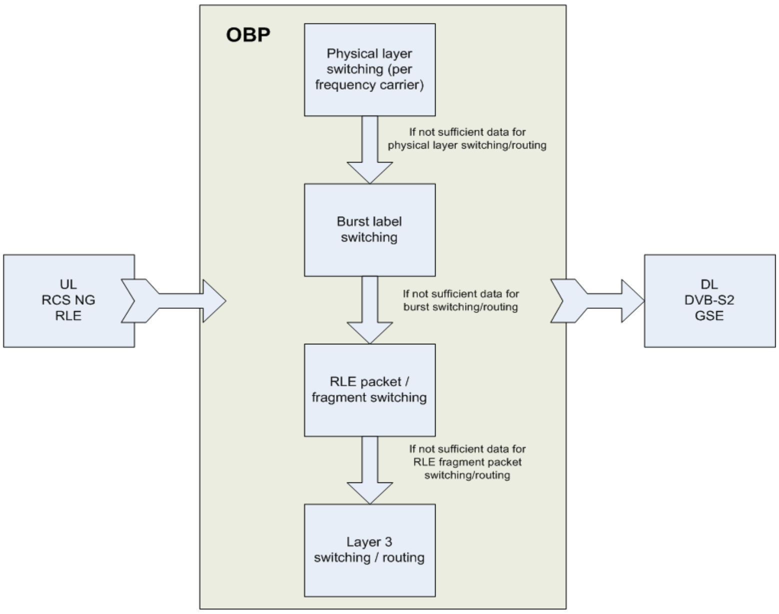

OBP
Physical layer switching (per frequency carrier)
If not sufficient data for physical layer switching/routing
Burst label switching
If not sufficient data for burst switching/routing
RLE packet / fragment switching
If not sufficient data for RLE fragment packet switching/routing
Layer 3 switching / routing
UL RCS NG RLE
DL DVB-S2 GSE

Figure 4.5: OBP switching / routing modes in RCS2

4.5 Encapsulation Scenarios

According to the payload architecture (regenerative or transparent) and the type of connectivity (star or mesh), there will be several possibilities for the encapsulation scheme of the different satellite links. These methods can be clarified with the help of the following scenarios:

- **Star-only** encapsulation scenario: this is the mandatory mode. RLE encapsulation is used in the return link and GSE in the DVB-S2 forward link. The RCST terminals compatible with this mode has an RLE encapsulator and a GSE decapsulator.

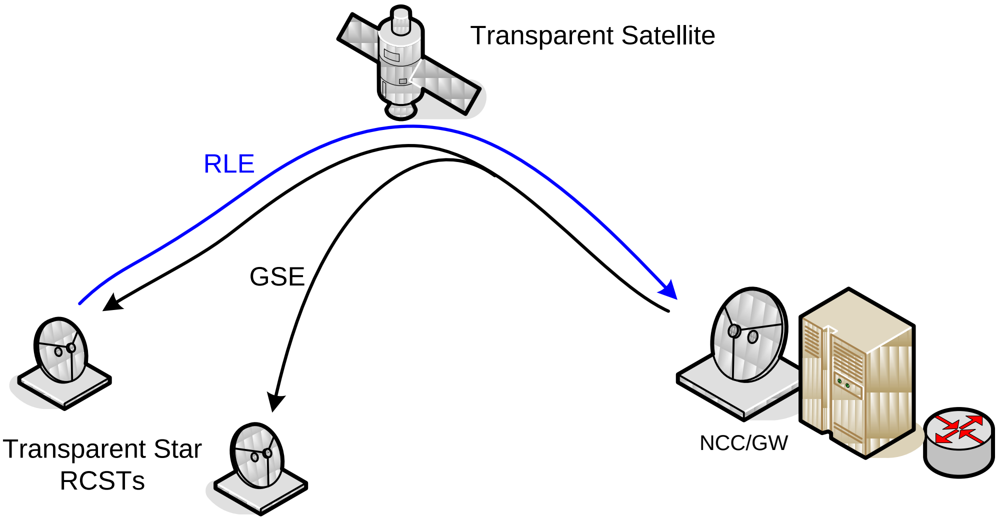


Figure 4.6: Star-only encapsulation scenario

- **Star and Mesh transparent encapsulation scenario**: this scenario is mandatory for mesh transparent systems. RLE encapsulation is present in the downlink due to the presence of mesh user terminals with DVB-RCS2 demodulator (in addition to the DVB-S2 demodulator). GSE is also in the downlink for reception of the forward link for signalling and star traffic. It is foreseen the presence of star-only and mesh terminals in the same satellite system. Mesh overlay RCSTs need two decapsulators, for RLE (user traffic) and GSE (DVB signalling) encapsulations.

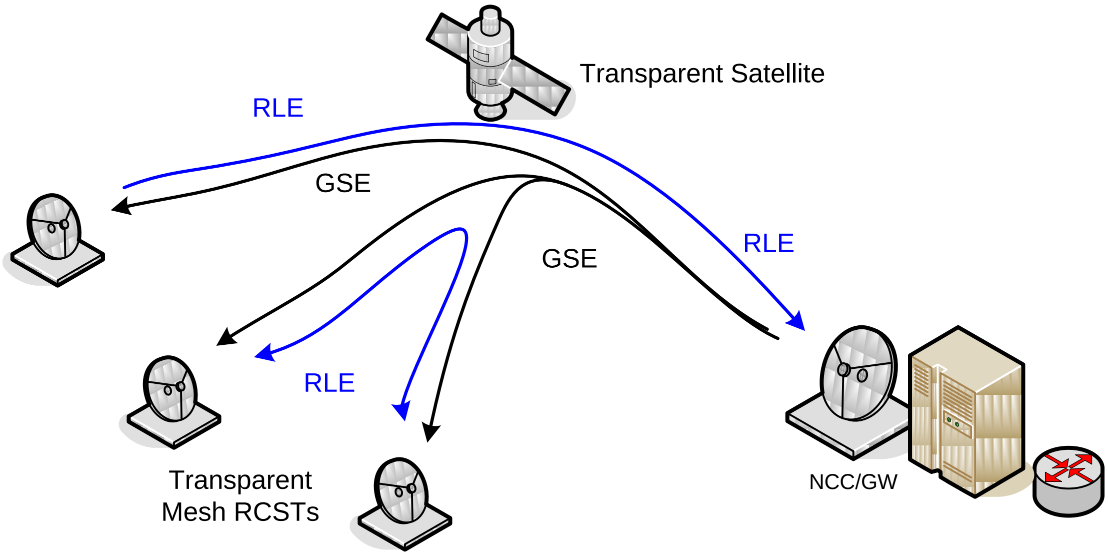


Figure 4.7: Star and Mesh transparent encapsulation scenario

- **Mesh regenerative (option 1)** encapsulation scenario: this scenario uses RLE for the uplink and GSE in the downlink, for all of the RCSTs and also for the NCC modem ODU (NCC-RCST). The OBP is in charge of translation of the encapsulation mode from RLE uplinks to GSE downlinks. This mode is intended for the use of standard RCSTs with little changes in SW (only related to DCP functionality) with regard to the transparent star terminals used in the star-only scenario.

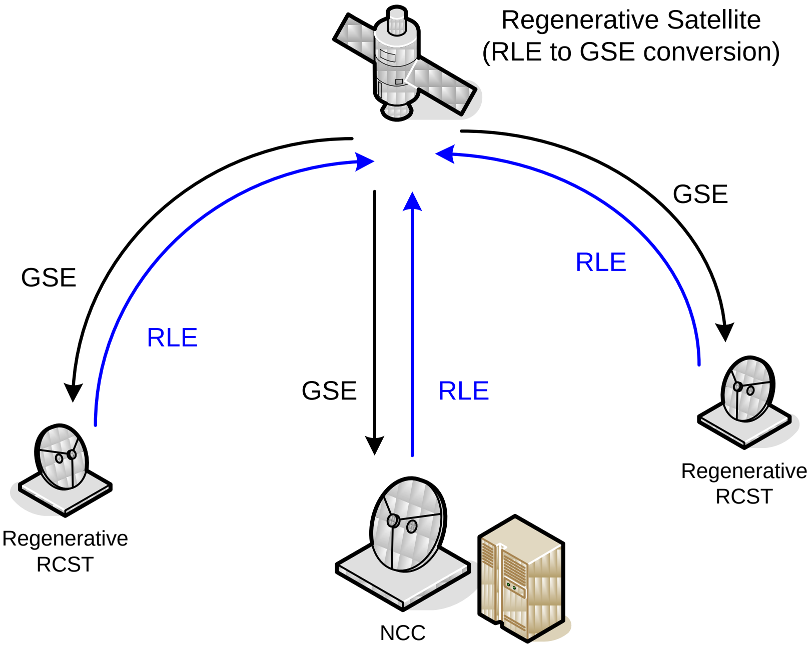


Figure 4.8: Mesh regenerative encapsulation scenario 1

- **Mesh regenerative (option 2)** encapsulation scenario: only RLE encapsulation is used, since the OBP does not perform any change in the encapsulation of uplink packets. The RCST supports RLE encapsulation and decapsulation functions, plus the DCP functionality for dynamic connectivity.

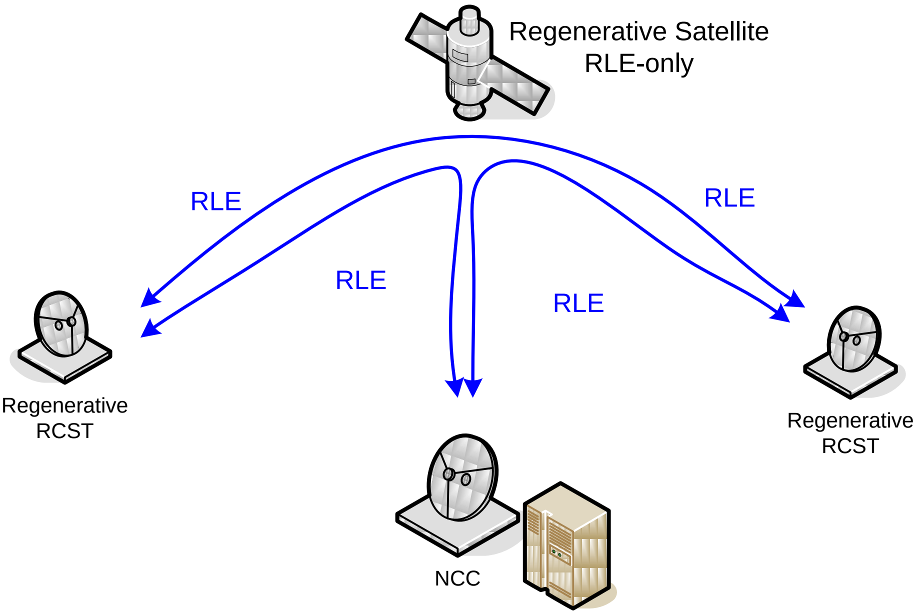


Figure 4.9: Mesh regenerative encapsulation scenario 2

4.6 Reference model for mobile Scenarios

A reference model for mobile DVB-RCS2 system architecture, as envisaged in the normative and system documents is depicted in Figure 4.10.

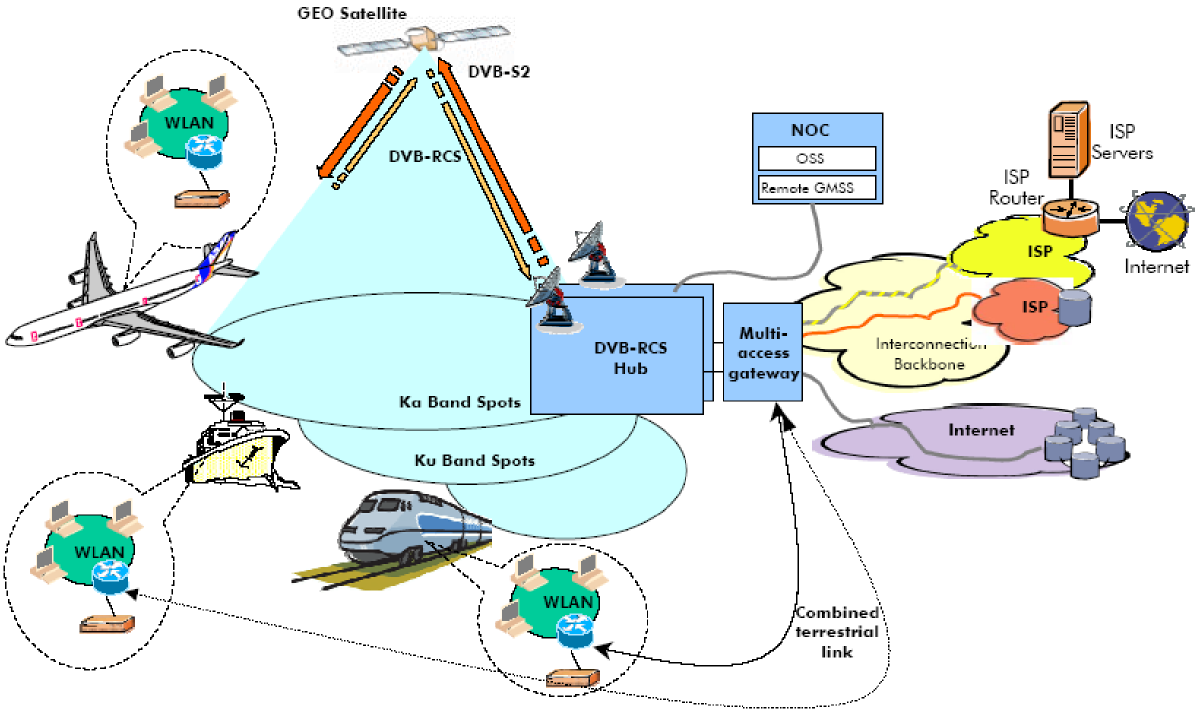


Figure 4.10: Overall system architecture for mobile interactive services via satellite

The space segment includes one or more GEO-satellites with a single or multi-beam configuration per satellite and with performances equivalent to those used for classical Ku-band or Ka-band fixed satellite services. Two service coverage scenarios are identified: one for regional case (one country or part of continent covered); and one for global case (land coverage complemented by oceans wide coverage, e.g. transatlantic).

Mobile RCSTs are in most cases mounted on a mobile platform operating as an access point for multiple users. The RF characteristics are adapted to the service requirements (in particular, antenna minimum/maximum size depending on the applications, regulatory and accommodations constraints). The IDU component, compliant to a mobile profile, covers the different applications.

The ground segment sub-system consists of the Network Operator Centre (NOC), the Network Control Centres (NCC) compatible with the definition of the NCC and the gateways, providing access to the terrestrial networks. The NOC is in charge of sharing the satellite resource among network operators, bandwidth allocation to NCC's and centralized management of satellite handover if relevant. The RCST is managed by one single NCC within the satellite coverage area.

Scenarios envisaged by a mobile system can be classified as follows:

- Line-Of-Sight (LOS) scenarios: these correspond to low-fading scenarios, which are almost always in LOS or close to LOS conditions. The aeronautical and maritime are the two main scenarios in this category.

- Non-LOS scenarios: these correspond to land-based strong-fading scenarios characterized by frequent/deep/long signal blockages and shadowing. The railway and the land vehicular are the two main scenarios in this category.

4.6.1 Line-Of-Sight Scenarios

4.6.1.1 Maritime

The maritime scenario comprises mainly: passenger transportation ships (like ferries and cruises); commercial ships (like cargos and tankers); and private transportation ships (like sailing boats).

Two coverage scenarios are particularly considered: a global one, corresponding to cruise routes (like transatlantic cruises); and a regional one, corresponding to ferry routes (for example, in Europe near the coast).

Regarding the modelling of the LOS channel conditions encountered in the maritime scenario, no experimental results concerning propagation measurements at the frequencies of interest are available in the literature. Nevertheless, considering the usage of high directive antennas, LOS conditions can be assumed and the channel can be regarded as a pure Ricean channel with very high Rice factor, i.e. very close to a purely AWGN channel.

4.6.1.2 Aeronautical

The aeronautical applications include providing telecommunication access to: passenger aircrafts (consisting in wide body and single aisle aircrafts); and private aircraft (like executive jets) or Unmanned Aeronautical Vehicles.

Two coverage scenarios are particularly considered: a global one, corresponding to long haul flights (like transatlantic flights); and a regional one, corresponding to short to medium haul flights (for example, over Europe).

For Line-of-Sight channel conditions encountered in the aeronautical scenario, experimental results at 18,6 GHz [i.48] show that the channel can be modelled as a Ricean during normal flight situations and manoeuvres, with a Rice factor well above 20 dB. Some signal fading, in the order of 3 dB, was observed for manoeuvres with roll angles up to 20°, whereas only in case of extreme manoeuvres, with roll angles up to 45°, the influence of the aircraft structure resulted in deep fades in the order of 15 dB. In conclusion, in aeronautical scenarios, LOS conditions can typically be assumed and the channel can be fairly approximated by a purely AWGN channel.

4.6.2 Non-LOS Scenarios

4.6.2.1 Railway

The mobility effects, such as multipath, shadowing and blockage, encountered due to the local environment in the vicinity of the mobile RCST, such as adjacent buildings, vegetation, bridges, and tunnels, result in non-LOS conditions of severe fading.

Several propagation measurements at Ku band [i.49] and Ka band [i.50], [i.51] were performed, based on which reference statistical channel models have been established:

- For Ku-band, the behaviour of the land mobile satellite channel can be modelled using a 3 state (namely LOS, shadowed and blocked) Markov chain based model, where each state is further characterized by a Rice distribution.

- For Ka-band, the behaviour of the land mobile satellite channel can be modelled using a 3 state (namely LOS, shadowed and blocked) Markov chain based model, where each state is further characterized by a Loo distribution.

The railroad satellite channel is in LOS state for most of the time. However, short blockages due to power arches as well as long blockages due to obstacles, such as buildings, vegetation, bridges, and tunnels, are also possible leading to non-LOS effects

4.6.2.2 Vehicular

The land vehicular scenario comprises mainly: passenger vehicles, commercial vehicles and private vehicles.

Only the regional coverage scenario is considered since vehicles remain within one continent.

For the modelling of the non-LOS channel conditions, the same statistical channel models described above for the railway scenario apply here as well except for deterministic and recurring signal attenuation (due to power arches for example) that is not applicable in non-LOS channel conditions.

5 Forward Link

Figure 5.1 shows one way to implement the forward link signalling from the NCC (the SI-information for RCS) to an existing DVB-S2 forward-link. The SI-tables with signalling data to the RCST from the NCC are represented as binary data (for example in binary files). These binary data are sent to the Gateway and put into a PSI/SI-inserter, together with the PSI/SI binary signalling data for DVB-S2.

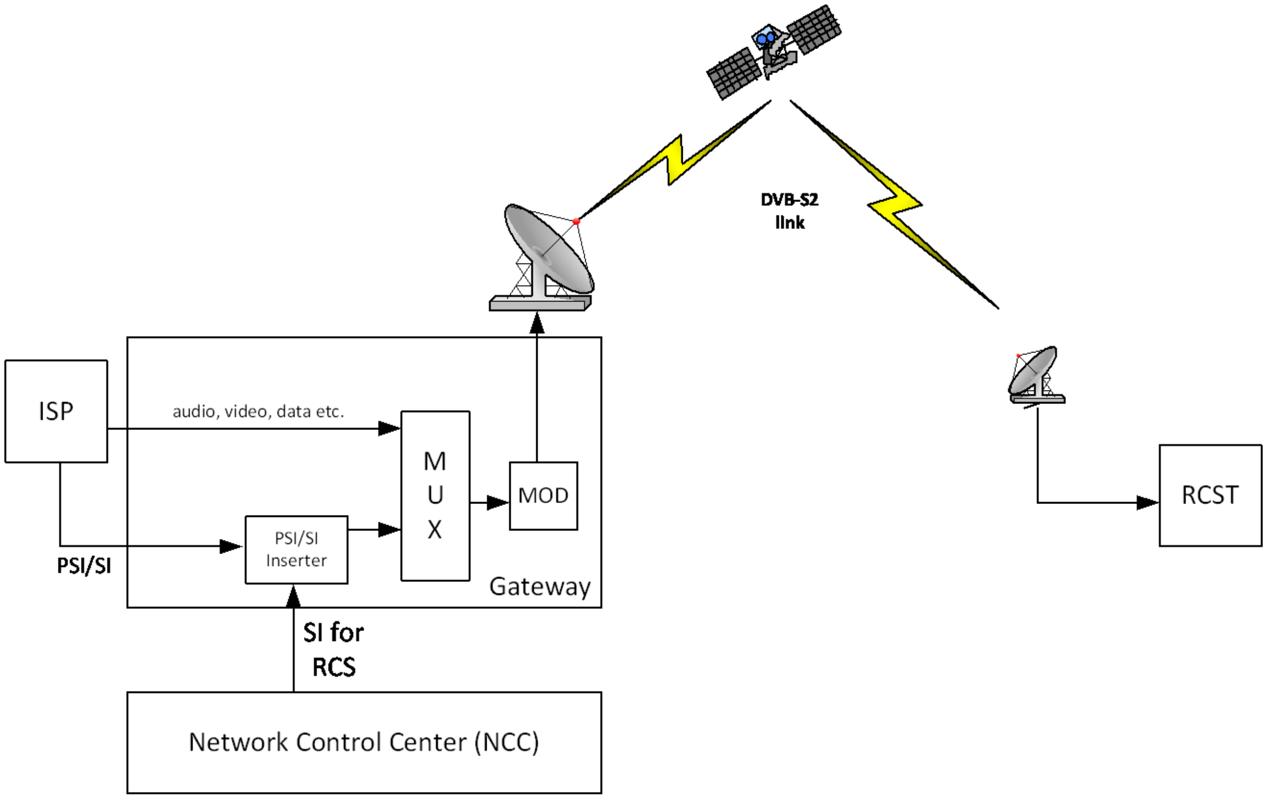


Figure 5.1: Implementation of SI signalling from NCC in DVB-S2

5.1 Forward Link Considerations

5.1.1 General Notes on ACM Operation

A DVB-S2/ACM forward link will typically be configured by the network operator with a "minimum" (most robust) and "maximum" (most spectrally efficient) MODCOD. The minimum MODCOD will be used to transmit critical data such as TBTPs and NCR synchronization. The selection of the minimum MODCOD will be based on the link budget analysis, and is chosen to ensure that all RCSTs are able to maintain forward link synchronization.

All RCSTs in the network will periodically relay their reception capability back to the hub using the return link data channel (see clause 8.3.2 of normative document [i.1]). The NCC will select a new MODCOD every time it transmits a BBFRAME.

The "ACM Gain" of the system is the effective spectral efficiency of the forward link divided by the spectral efficiency that would have been achieved if only the "minimum" MODCOD had been used.

In order to maximize the ACM Gain of the system, the NCC should attempt to send all unicast data to RCSTs using their best achievable MODCOD. To do this, the NCC needs to maintain a table of the current MODCOD reception capability of each RCST in the network.

The system should allow the MODCOD to potentially change with every BBFRAME to maximize the ACM Gain.

It is important to note that in a DVB-S2/ACM system, any given RCST may be incapable of receiving many/most of the BBFRAMEs on the forward link. This depends on the location of the RCST within the footprint of the satellite beam.

5.1.2 FEC Frame Size Selection

In DVB-S2/ACM, there is a fundamental trade-off between the increased coding gain achieved by normal size DVB-S2 frames, and the reduced latency achieved by short DVB-S2 frames.

For many interactive applications, reducing the latency with short frames is more important than the 0,2 - 0,3 dB in coding gain (see clause 6 in [i.2]) with normal frames at a given MODCOD. The small loss of efficiency can usually be recovered with fine-grained MODCOD selection using ACM as described in clause 5.1.1.

Experience has shown that there is a performance gain when using short frames, compared to normal frames, given the same target latency for data packets. Normal frames with short target latencies result in many large BBFRAMEs being transmitted with padding, thus wasting bandwidth.

5.1.3 Receiver Operation

A non-viable MODCOD is one that the receiver knows it cannot demodulate because the signal quality, measured in signal to noise plus interference ratio (SNIR), of the received signal is too low. A background software control loop in the RCST can periodically determine the SNIR, and set the demodulator to filter out these MODCODs. This prevents known-bad input from entering the higher layers. Depending on the demodulator implementation, it can also provide an opportunity for the LDPC decoder to perform more iterations on the viable MODCODs.

RCSTs should attempt to decode data on all viable MODCODs. There is no explicit signalling from the hub to indicate which MODCODs a terminal should demodulate or which MODCOD the NCC considers as the lowest viable MODCOD for the RCST.

5.1.4 Pilot Symbols

When using ACM, the demodulator can only rely on the presence of PLHEADER and pilot symbols. Terminals at the edge of a beam may not have many viable MODCODs to detect. The use of time interval between PLHEADERs alone can be insufficient to allow the RCST to remain locked to the DVB-S2 signal.

Frames with pilot symbols have negligible overhead (see clause 5.5 of [i.2]). The efficiency ranges from 99,3 % to 99,7 % when using normal frames; 97,3 % to 98,9 % when using short frames.

It is therefore recommended that pilot symbols always be used for DVB-S2/ACM deployments.

5.1.5 Restrictions on Transmitting DVB-S2 Dummy Frames

Some existing DVB-S2 demodulator ASIC implementations could dysfunction when DVB-S2 signals contain dummy frames, even though such chipsets are otherwise ACM-capable.

As a work-around for the above mentioned issue, it is recommended that empty (DFL=0) 32APSK short frames, with pilots, be used in lieu of dummy frames.

A DVB-S2 dummy frame is 36 SLOTs, plus 1 SLOT for the PLHEADER (37 x 90 = 3 330 symbols). This is the same length as a short 32-APSK frame with no pilot symbols. A short 32APSK frame with pilots is 3 402 symbols, an increase of 2 %.

5.1.6 GSE Fragmentation

A data packet transmitted using GSE may be fragmented prior to transmission. The resulting fragments may be transmitted over multiple BBFRAMEs. The system should allow GSE fragments to be transmitted on BBFRAMEs of any viable MODCOD that the (set of) target RCSTs is capable of receiving. This helps to improve the overall ACM gain of the system.

5.1.7 Explicit Integrity Protection of SDU

Clause 5.1.1.2 of the normative document [i.1] indicates that the RCST supports a modified format and syntax of the BBFRAME data field for transport of the GSE PDUs by inserting a CRC32 as the last four bytes of the BBFRAME data field. A background and rationale for such protection is provided below.

5.1.7.1 Background

The GSE protocol relies on error free reception of PDU's contained in each BBFRAME for proper GSE operation [i.3]. While this is a valid assumption for certain scenarios, such as CCM transmission to fixed receivers in DVB-S2 broadcast systems, it may not hold in other circumstances that are important for interactive services. In particular, transient propagation conditions in ACM operation and the occurrence of signal interruptions, which are common in land-mobile scenarios, are examples of situations where further error detection capability may be required. These two cases are further analysed below.

ACM Systems

For continued, error-free operation ACM systems need to include a margin that allows reception while a MODCOD change is being carried out. In order to maximize the capacity, it is of course desirable to minimize this margin. The margin needs to account for *C/(N+I)* measurement uncertainty as well as for additional fade that occurs in the switchover latency interval.

In the following analysis, an undetected frame error probability of 10^{-7} (quasi error free) is targeted. For the worst-case BCH detection performance of 2 x 10^{-4}, the actual frame error probability should therefore be kept below 5x10^{-4} at all times. When relying on the BCH decoder, the link margin should therefore ensure that the probability of a frame being received below the actual detection threshold for the current MODCOD is smaller than this value.

A typical RMS C/(N+I) measurement uncertainty is 0,1 dB. This can be achieved relatively quickly even at low C/(N+I), for example by data-aided estimation on approximately 4 000 symbols. This corresponds to the frame headers and pilots of about 13 short QPSK frames. Assuming a Gaussian distribution (which is reasonable for a data-aided estimate), the error magnitude exceeded with a probability of 5 x 10^{-4} is ~3,5 σ or 0,35 dB.

Fading during the system reaction time depends strongly on the current fade level as well as on implementation details. A model of the statistical distribution of fade slope is given in [i.35]. This model predicts the probability that a given fade slope (in dB/s) is exceeded for a given level of fading. Figure 5.2 is computed according to that model and shows the fade slope exceeded for 0,05 % of the time (= 5 x 10^{-4}) as a function of the instantaneous fade. It can be noted that the model is frequency-independent; propagation measurements have shown a very similar behaviour in all bands. However, the probability of actually having a fade of a given level of course depends on the band.

The system reaction time is very implementation-specific. The physical lower bound is one round-trip time (500 ms).

The total margin needed is the sum of the measurement uncertainty and the worst-case additional fade in the reaction interval. Assuming statistical independence of the two terms, we use the values for 5 x 10^{-4} for each to predict the overall value for the same probability.

For a 1-second reaction time near clear weather (2 dB fade), the margin is 0,44 dB. For a 10 dB fade and 5 seconds reaction time, it is 2,7 dB. This corresponds to a significant loss of capacity, so it is highly desirable to operate with smaller margins. This however increases the probability of errored frames during the transient. While some frame errors can sometimes be tolerated, it is important that the errors are detected reliably.

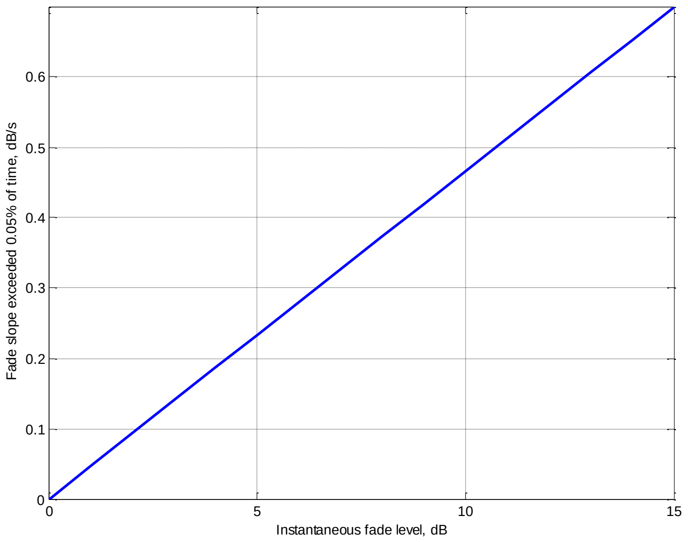


Figure 5.2: Fade slope exceeded 0,05 % of time

Land-Mobile Systems

Signal blockages in non-line-of-sight mobile (NLOS) systems are equivalent to fades that occur very rapidly. A blockage can occur at any time. Depending on the receiver implementation, it is quite possible that the frame may have been detected before the loss of signal occurred. In other words, the on-set of a signal blockage is likely to be seen as a frame error. It is important for the operation of the link-layer FEC NLOS countermeasures that these frame errors are detected properly so that the corresponding symbols can be marked as erasures for the LL-FEC decoders.

Consider a high-speed train moving at 350 km/hr (97 m/s). With power poles spaced at 50 m, there will be 1,94 blockages/s just from these. Further assume that each blockage causes one detected-but-errored frame and that the probability of undetected frame errors is given by the performance of the BCH decoder (2 x 10^{-4} = 1 in 5 000, worst case). This will result in an undetected frame error on average every 5 000 / 1,94 seconds or 43 minutes. This is clearly inadequate.

Experimental results presented confirm that the error detection capability of the BCH decoder is not sufficient to achieve the desired robustness. Analytical results however, confirm that an explicit 32-bit CRC check will provide the necessary robustness.

In order to allow the system implementer the flexibility to trade off the margin against the risk of frame errors during the transient, it is recommended that the encapsulation should be robust against such frame errors; in particular since this can be achieved with the minimal overhead incurred by a CRC-32 BBFRAME carrying GSE PDUs as shown in Figure 5.3.

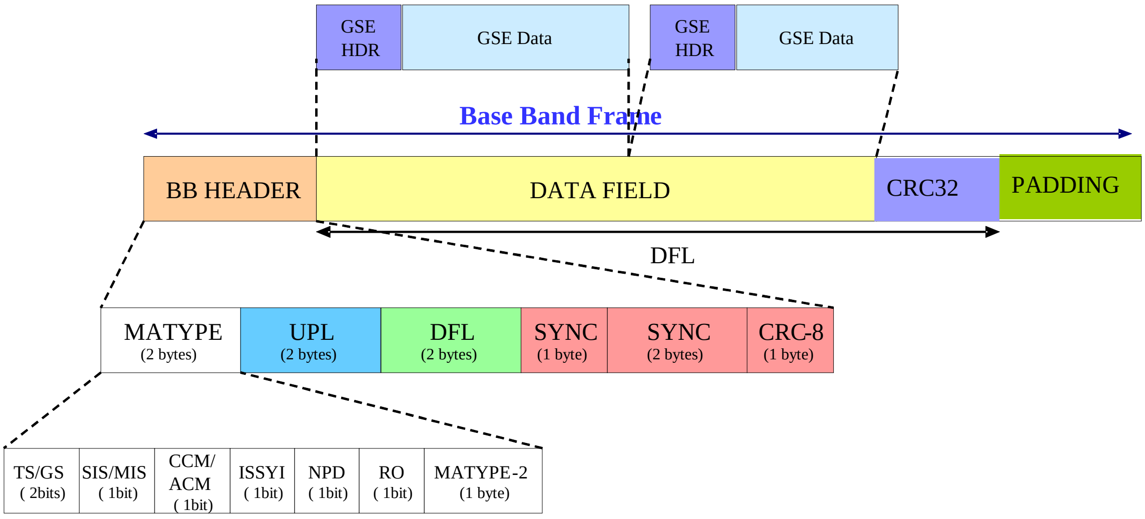


Figure 5.3: Recommended use of CRC32 per BBFRAME carrying GSE PDUs

5.2 Forward Link L2S Considerations

5.2.1 L2S Insertion

Most broadcast L2S messages are fundamental to RCS2 system operation, and should therefore be transmitted on the lowest system MODCOD. These messages include: NIT, SCT, FCT2, BCT, SPT and TBTP2.

Unicast L2S messages may be transmitted on any MODCOD that can be received by the targeted RCST.

5.2.2 NCR Insertion

It is recommended that NCR timestamps be inserted at a fixed, periodic interval. An RCST can detect the transmission interval and optimize its NCR tracking loop accordingly. This is not a hard-realtime requirement; however, the NCC should aim to transmit NCR timestamps at a reasonable fixed interval.

The normative document specifies as a minimum the NCR Insertion rate of 10 times per second (see clause 6.3 of [i.1]). In order to minimize the impact of the L2S on the forward link overhead it is desirable to maintain the NCR insertion rate close to the minimum. However, due to operation and channel conditions, such as support of mobile services, it may be necessary to use a higher NCR insertion rates. In certain system scenarios NCR transmission rates of up to 32 times per second has been reported. It is therefore recommended to carefully trade off desired efficiency with the required frame loss resilience in the operational environment when selecting the NCR insertion rate.

Because the NCR is fundamental to RCS2 system timing, it should always be transmitted on the lowest system MODCOD. The NCR transmission frequency will therefore affect the overall ACM gain achievable in the system.

5.2.3 FL L2S Components

The normative document (See clause 6 of [i.1]) provides the description and use of lower layer signalling components. In this section, some guidelines on the use of L2S components are provided.

5.2.3.1 NIT

It is recommended that the following descriptors are present in NIT:

- Linkage descriptor (for finding the RCS service information)
- Multiplex Stream Specification containing:
  - System delivery descriptor
  - S2 System delivery descriptor

5.2.3.2 SCT/FCT2/BCT/TBTP2

It is assumed that the SCT, FCT2 and BCT contents do not change frequently, so the issues related to handling of changing superframe configurations dynamically are considered outside the scope of the guidelines.

The TBTP2 is used to allocate time slot allocations to single (DAMA) or groups (RA) of terminals. The flexibility in the TBTP2 allows dynamic superframe to superframe changes of the frame allocation including changes in:

- MODCOD
- Symbol rate
- Burst duration changes (by aggregating continuous allocated BTUs
- Access method (DAMA or RA)
- Burst type content (Traffic, signalling or both)

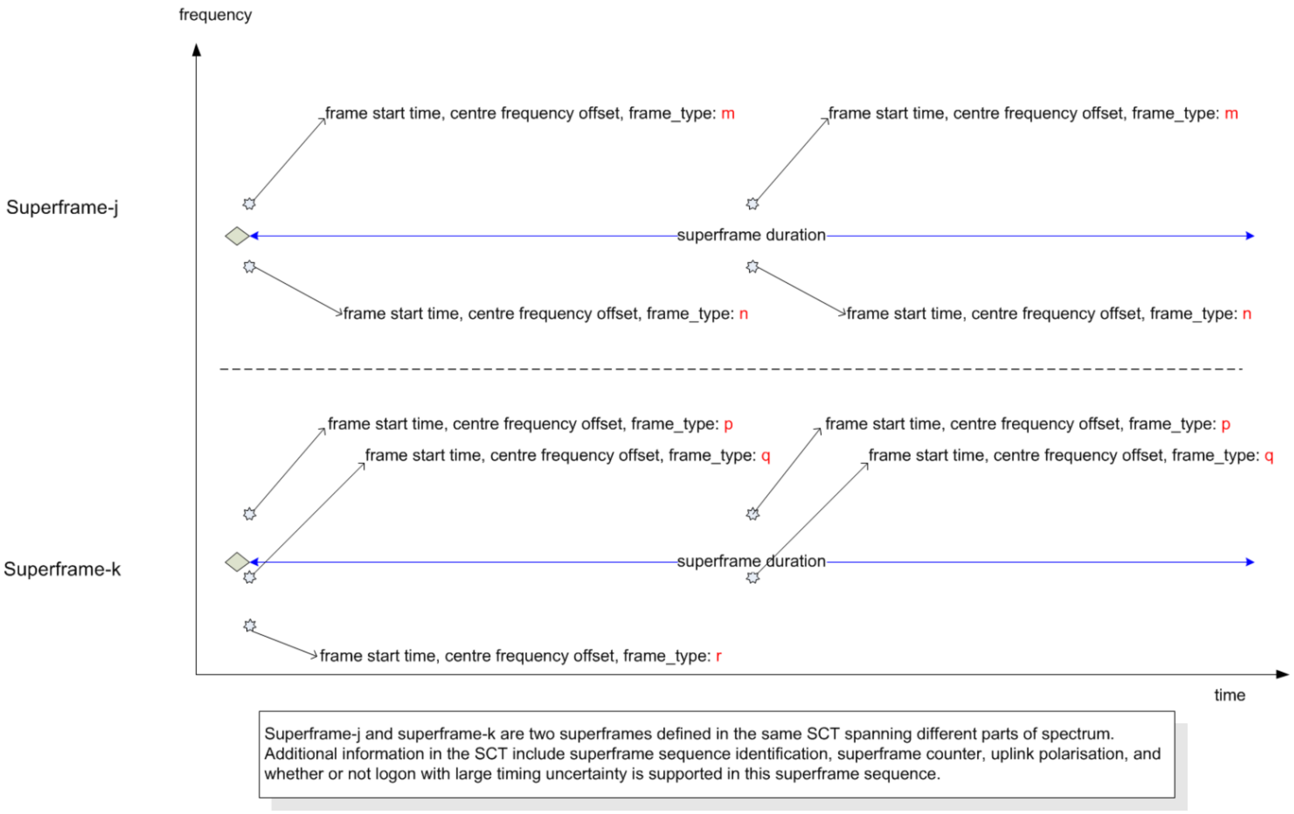
It should be noted that the frame duration in FCT2 has the format of NCR counter as integer, while the same value in the FCT from [i.1] used the PCR base and extension format.

Figure 5.4 shows across time and frequency dimensions the information included in the SCT. Two superframes are included in the SCT in this example, covering different portions of the frequency. It should be emphasized that this illustration deliberately shows the information that can be deduced from the SCT only. The complete definition of superframes will need joint interpretation of SCT, FCT2, BCT, and TBTP2 tables. Primarily, the superframe definition within the SCT provides information regarding:

- the start time, the duration, and the centre frequency of a superframe;
- the start time and the centre frequency for each frame within the superframe.

Each frame in the superframe is identified by an implicit frame numbering within the superframe, each frame referring to a frame type specifying its structure. FCT2 contains the specification of each frame type.

As stated in the diagram, the SCT also contains information regarding the superframe sequence identification, superframe counter, uplink polarization, and whether or not logon with large timing uncertainty is supported in this superframe sequence. Each superframe sequence has its own superframe specification in the SCT.


frequency
Superframe-j
frame start time, centre frequency offset, frame_type: m
frame start time, centre frequency offset, frame_type: m
superframe duration
frame start time, centre frequency offset, frame_type: n
frame start time, centre frequency offset, frame_type: n
Superframe-k
frame start time, centre frequency offset, frame_type: p
frame start time, centre frequency offset, frame_type: q
frame start time, centre frequency offset, frame_type: p
frame start time, centre frequency offset, frame_type: q
superframe duration
frame start time, centre frequency offset, frame_type: r
time

Superframe-j and superframe-k are two superframes defined in the same SCT spanning different parts of spectrum. Additional information in the SCT include superframe sequence identification, superframe counter, uplink polarisation, and whether or not logon with large timing uncertainty is supported in this superframe sequence.

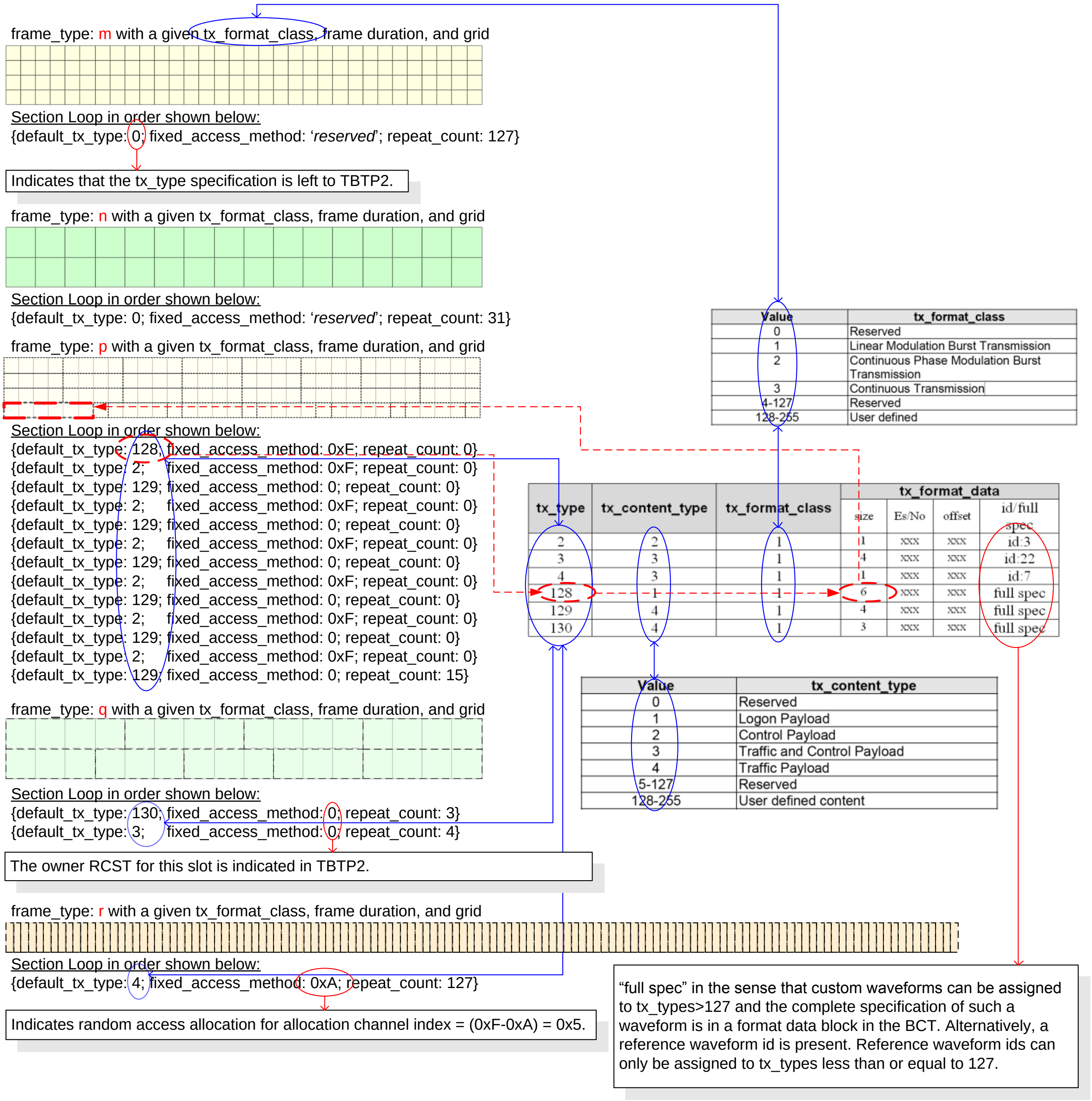
Figure 5.4: Visualization of SCT Content

Figure 5.5 shows five different frame type examples. The BTU grid, which is completely specified in the FCT2, is pictorially shown for each frame type. In addition, FCT2 may contain a section loop for each frame type. The red dashed arrows in Figure 5.5 presents an example timeslot definition within the frame type, p. The first section in the section loop corresponds to the first timeslot in the frame (lowest frequency, lowest start time). These red dashed arrows show the correct order in which different elements in the FCT2 should be read to define the first timeslot, which spans 6 BTUs in the lowest frequency and first BTU in time. The example tx_type is 128. Also shown in the figure are the tx_content_type and tx_format_class that correspond to tx_type:128; this information would be available in the BCT. For tx_type:128, the example shows a tx_content_type:1 and a tx_format_class:1. As shown in the tables in the example, tx_content_type:1 indicates that the timeslot is to be used for logon and tx_format_class:1 indicates that the timeslot is to carry a Linear Modulation burst. Back in the section loop, the fixed_access_method parameter for this timeslot is 0xF, which indicates that the timeslot is for random access. Had the fixed_access_method been 0 for this timeslot, then the timeslot would have been a dedicated access timeslot, the owner which would have been indicated in a follow-up TBTP2 table.

The frame type may combine a number of consecutive BTUs into a timeslot. For each such combination, FCT2 may indicate a `default_tx_type` and a `fixed_access_method`. The `default_tx_type` =0 indicates that the tx_type is determined in TBTP2. Also the non-zero `default_tx_type` allocations may be overridden by a `tx_type` specified for the timeslot in TBTP2.

The transmission in a timeslot with a given `tx_type` is completely determined by the `tx_type` specification contained in the BCT table. The figure below shows a simplified BCT example to illustrate how FCT2 and BCT jointly specify the stationary structure of each frame type. The BCT determines the timeslot size for each `tx_type` in BTU units. Thus, the timeslot duration in absolute time depends on the BTU specification. Timeslots using the same `tx_type` may have different duration when appearing in frames of different frame types.

Figure 5.6 completes superframes that would emerge by joint interpretation of the example SCT, FCT2, and BCT contents.



frame_type: m with a given tx_format_class, frame duration, and grid

Section Loop in order shown below:
{default_tx_type: 0; fixed_access_method: '*reserved*'; repeat_count: 127}

Indicates that the tx_type specification is left to TBTP2.

frame_type: n with a given tx_format_class, frame duration, and grid

Section Loop in order shown below:
{default_tx_type: 0; fixed_access_method: '*reserved*'; repeat_count: 31}

frame_type: p with a given tx_format_class, frame duration, and grid

Section Loop in order shown below:
{default_tx_type: 128; fixed_access_method: 0xF; repeat_count: 0}
{default_tx_type: 2; fixed_access_method: 0xF; repeat_count: 0}
{default_tx_type: 129; fixed_access_method: 0; repeat_count: 0}
{default_tx_type: 2; fixed_access_method: 0xF; repeat_count: 0}
{default_tx_type: 129; fixed_access_method: 0; repeat_count: 0}
{default_tx_type: 2; fixed_access_method: 0xF; repeat_count: 0}
{default_tx_type: 129; fixed_access_method: 0; repeat_count: 0}
{default_tx_type: 2; fixed_access_method: 0xF; repeat_count: 0}
{default_tx_type: 129; fixed_access_method: 0; repeat_count: 0}
{default_tx_type: 2; fixed_access_method: 0xF; repeat_count: 0}
{default_tx_type: 129; fixed_access_method: 0; repeat_count: 0}
{default_tx_type: 2; fixed_access_method: 0xF; repeat_count: 0}
{default_tx_type: 129; fixed_access_method: 0; repeat_count: 15}

frame_type: q with a given tx_format_class, frame duration, and grid

Section Loop in order shown below:
{default_tx_type: 130; fixed_access_method: 0; repeat_count: 3}
{default_tx_type: 3; fixed_access_method: 0; repeat_count: 4}

The owner RCST for this slot is indicated in TBTP2.

frame_type: r with a given tx_format_class, frame duration, and grid

Section Loop in order shown below:
{default_tx_type: 4; fixed_access_method: 0xA; repeat_count: 127}

Indicates random access allocation for allocation channel index = (0xF-0xA) = 0x5.

Value	tx_format_class
0	Reserved
1	Linear Modulation Burst Transmission
2	Continuous Phase Modulation Burst Transmission
3	Continuous Transmission
4-127	Reserved
128-255	User defined

tx_type	tx_content_type	tx_format_class	tx_format_data			
			size	Es/No	offset	id/full spec
2	2	1	1	xxx	xxx	id:3
3	3	1	4	xxx	xxx	id:22
4	3	1	1	xxx	xxx	id:7
128	1	1	6	xxx	xxx	full spec
129	4	1	4	xxx	xxx	full spec
130	4	1	3	xxx	xxx	full spec

Value	tx_content_type
0	Reserved
1	Logon Payload
2	Control Payload
3	Traffic and Control Payload
4	Traffic Payload
5-127	Reserved
128-255	User defined content

"full spec" in the sense that custom waveforms can be assigned to tx_types>127 and the complete specification of such a waveform is in a format data block in the BCT. Alternatively, a reference waveform id is present. Reference waveform ids can only be assigned to tx_types less than or equal to 127.

Figure 5.5: Visualization of FCT2 Content

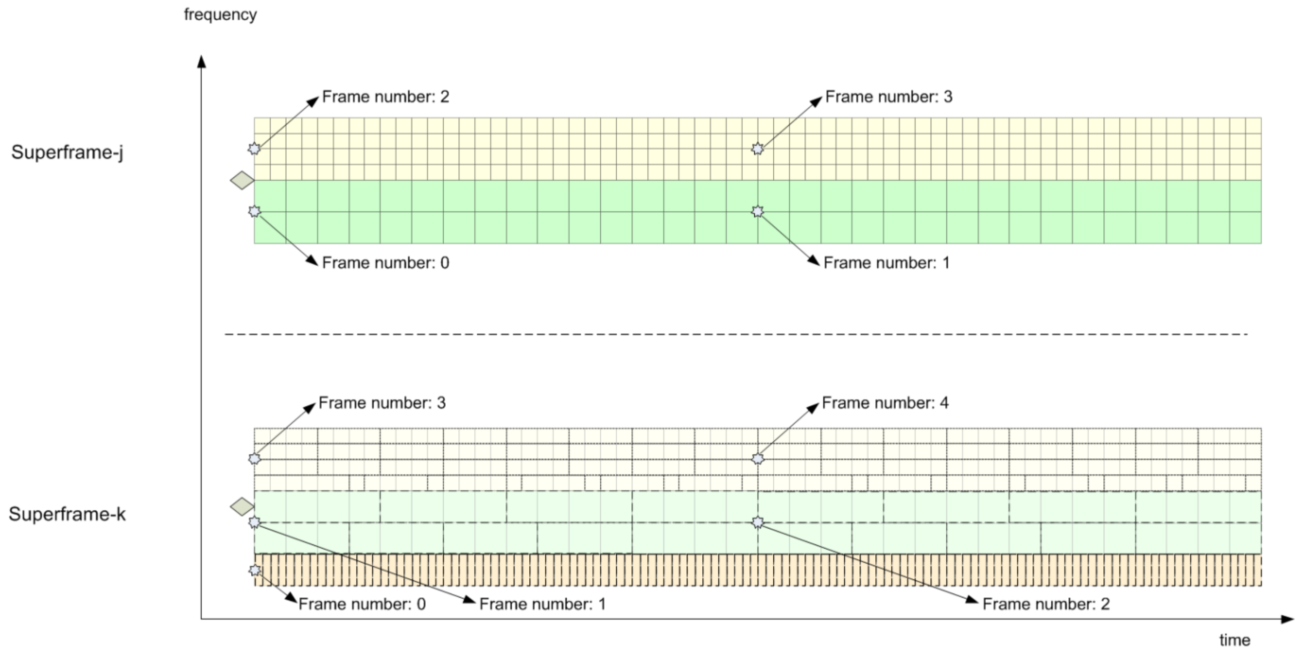
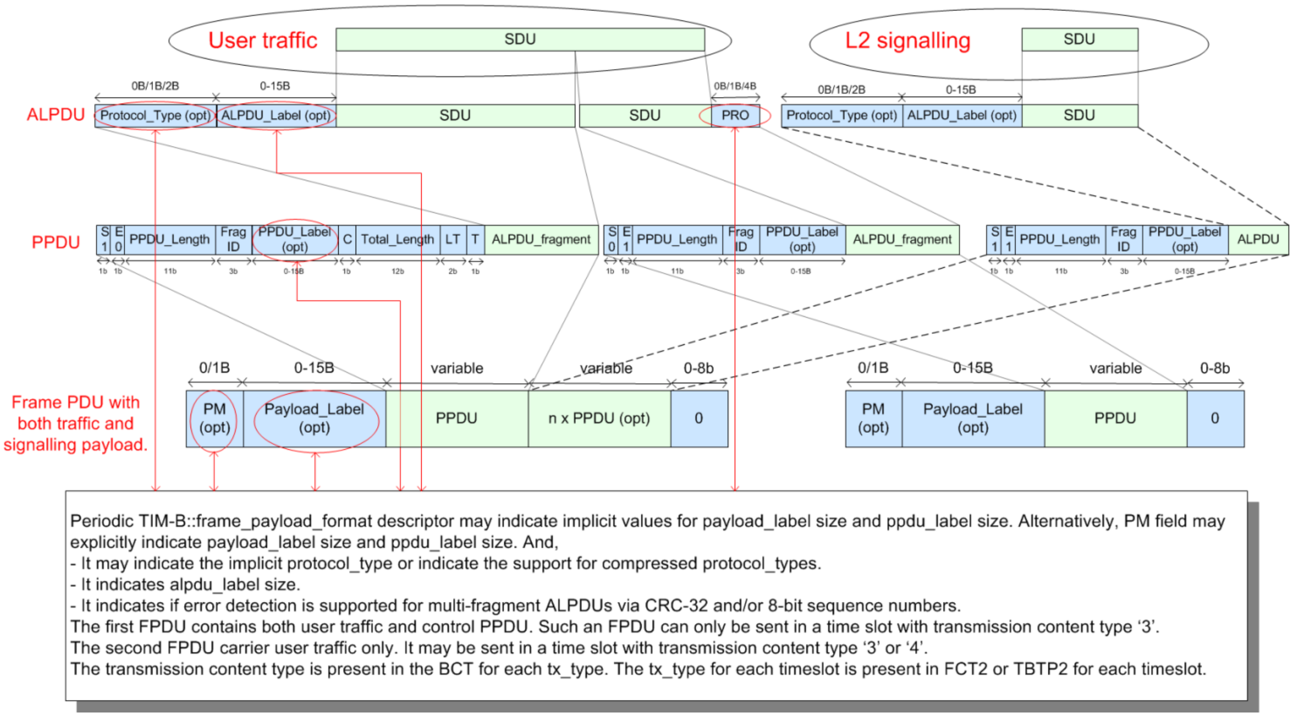


Figure 5.6: Superframe Composition, as specified in SCT, FCT2, BCT

Figure 5.7 shows an example encapsulation of single user traffic SDU and one L2 signalling SDU in two FPDUs. The first FPDU is transmitted in a timeslot that is associated with a `tx_content_type`: '3' in the BCT (see Figure 5.5 for tx_content_types), since it carries both user traffic and L2 signalling. The second FPDU is sent in a timeslot with `tx_content_type`: '3' or '4', since it only carries user traffic. Thus, the encapsulation procedure should take into account the content type allowed for a given timeslot before building the payload.



Periodic TIM-B::frame_payload_format descriptor may indicate implicit values for payload_label size and ppdu_label size. Alternatively, PM field may explicitly indicate payload_label size and ppdu_label size. And,
- It may indicate the implicit protocol_type or indicate the support for compressed protocol_types.
- It indicates alpdu_label size.
- It indicates if error detection is supported for multi-fragment ALPDUs via CRC-32 and/or 8-bit sequence numbers.

The first FPDU contains both user traffic and control PPDU. Such an FPDU can only be sent in a time slot with transmission content type ‘3’.
The second FPDU carrier user traffic only. It may be sent in a time slot with transmission content type ‘3’ or ‘4’.
The transmission content type is present in the BCT for each tx_type. The tx_type for each timeslot is present in FCT2 or TBTP2 for each timeslot.

Figure 5.7: User Traffic SDU Encapsulation

Timeslots that only carry L2 signalling are associated with `tx_content_type` '1' (for Logon) or '2' (for Control) in the BCT. Figure 5.8 shows an example of L2 signalling being encapsulated in a FPDU. With such timeslots, the L2 signalling SDU need not ALPDU and PPDU headers.

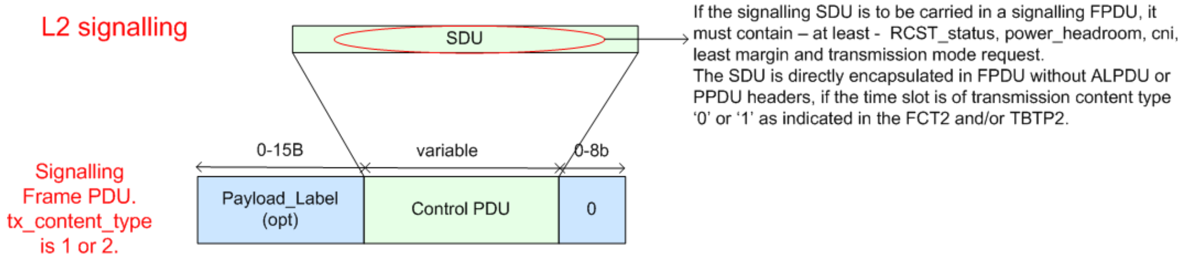

L2 signalling
SDU
If the signalling SDU is to be carried in a signalling FPDU, it must contain – at least - RCST_status, power_headroom, cni, least margin and transmission mode request.
The SDU is directly encapsulated in FPDU without ALPDU or PPDU headers, if the time slot is of transmission content type '0' or '1' as indicated in the FCT2 and/or TBTP2.
0-15B
variable
0-8b
Signalling Frame PDU. tx_content_type is 1 or 2.
Payload_Label (opt)
Control PDU
0

Figure 5.8: L2 Signalling Encapsulation

It should be noted that the SCT/FCT2/BCT/TBTP2 are sufficient to identify RA timeslots in a superframe.

RA timeslots with tx_content_type '1'or '2' are for L2 signalling only, and Slotted ALOHA is the random access mechanism to be used with these timeslots.

RA timeslots with tx_content_type '3' or '4' may carry user traffic. The specific access mechanism for these slots (Slotted ALOHA vs. CRDSA) is indicated in periodic Random Access Traffic Method descriptors.

6 Return Link

6.1 Fixed-RCST Physical Layer Synchronization

For administrative and technical reasons the location of an RCST providing fixed satellite services should be known to the network operator.

An RCS system can be designed assuming an accuracy of the location (longitude, latitude and altitude) of the RCST of no more than a few kilometers. Some network operators may require a better accuracy.

It is recommended that commonly available high precision localization systems be used during the RCST installation or any re-installation.

If possible, the NCC should correct for satellite translation error and Doppler shift introduced on the NCC-to-Satellite uplink and the Satellite-to-NCC downlink. The residual frequency offset between any two RCSTs includes effects due to Doppler shift on the Satellite-to-RCST downlink and the RCST-to-Satellite uplink. The residual relative frequency offset needs also to be compensated for by the NCC.

6.1.1 DVB S2 Receiver implementation aspects

There is a requirement to associate, without any ambiguity, the detected SOF with decoded frames (and hence with NCR fields within the frame), and to make this information available to the NCR synchronization circuit in the RCST. Decoding processing delays are far less critical.

An association and interfacing approach that puts almost no functional requirements on the DVB S2 RX chip is outlined in DVB-S2 Specifications, annex G5 [i.2].

Decoding delay jitter does not degrade the network clock reference extracted by the terminal. It is important however to verify that SOF detect circuit has low jitter. (Typical requirement, depending on NCR tracking design: SOF detect jitter < 100 ns pp).

It is recommended that systematic delays in the SOF detect circuit (for example the delay of the RX filter) are documented by the DVB S2 RX circuit provider. These delays can then be compensated for in the RCST design, if needed.

It is also recommended to prevent that data flagged as unreliable enter the NCR tracking circuit. Unreliable data flags can originate from in a DVB S2 receiver from BBHEADER checking, BCH decoding checks, or the CRC-32 associated per BBFRAME (as per clause 5.1.1.2 of the normative document [i.1]).

6.2 Mobile RCST Synchronization

The present clause addresses the impact of mobility (i.e. Doppler) on the physical layer performances on the return link.

6.2.1 Doppler shift and time drift

Table 6.1 gives some typical values in Ku-band for Doppler shift and time drift, for different types of mobile terminals (i.e. with various speed and acceleration). The table provides worst case Doppler shifts, assuming terminal motion towards the satellite and minimum elevation angle (leading to a minimum relative angle θ between the vehicle and the satellite θ=0). The Doppler values due to satellite motion are also included for reference.

Table 6.1: Doppler shift in Ku-band for different types of mobile terminals

Type of mobile terminal (note 1)	Speed	Acceleration (m/s^2)	Doppler rate (note 2)	Uplink Doppler frequency shift (note 3) (Hz)	Downlink Doppler frequency shift (note 4) (Hz)	Time drift (ns/s)	Uplink frequency drift (Hz/s)	Downlink frequency drift (Hz/s)
Pedestrian	5 km/h	1	4,6E-09	67	59	4,6	48	43
Maritime	25 km/h	5	2,3E-08	336	295	23,1	242	213
Vehicular	120 km/h	10	1,1E-07	1 611	1 417	111	483	425
Train	350 km/h	5	3,2E-07	4 699	4 132	324	242	213
Aeronautical	330 m/s	17	1,1E-06	15 950	14 025	1 100	822	723
Satellite	3 m/s	0	1,0E-08	145	128	10	4,8	4,3

NOTE 1: Vehicular: bus, car, truck
Aeronautical: < speed of sound
Satellite: satellite movement (GSO) assuming satellite motion is versus nadir (reference point)

NOTE 2: The maximum Doppler values due to satellite motion are typical for geostationary satellite during main mission life. The worst case Doppler values (e.g. when satellite mission is extended using inclined orbit satellite) are not considered here.

NOTE 3: Uplink frequency: 14,5 GHz

NOTE 4: Downlink frequency: 12,75 GHz

Table 6.2 provides typical values for Doppler shift in Ka-band.

Table 6.2: Doppler shift in Ka-band for different types of mobile terminals

Type of mobile terminal (note 1)	Speed	Acceleration (m/s^2)	Doppler rate (note 2)	Uplink Doppler frequency shift (note 3) (Hz)	Downlink Doppler frequency shift (note 4) (Hz)	Time drift (ns/s)	Uplink frequency drift (Hz/s)	Downlink frequency drift (Hz/s)
Pedestrian	5 km/h	1	4,6E-09	139	94	4,6	100	67
Maritime	25 km/h	5	2,3E-08	694	468	23,1	500	337
Vehicular	120 km/h	10	1,1E-07	3 333	2 244	111	1 000	673
Train	350 km/h	5	3,2E-07	9 722	6 546	324	500	337
Aeronautical	330 m/s	17	1,1E-06	33 000	22 220	1 100	1 700	1 145
Satellite	3 m/s	0	1,0E-08	300	202	10	10,0	6,7

NOTE 1: Vehicular: bus, car, truck
Aeronautical: < speed of sound
Satellite: satellite movement (GSO) assuming satellite motion is versus nadir (reference point)

NOTE 2: The maximum Doppler values due to satellite motion are typical for geostationary satellite during main mission life. The worst case Doppler values (e.g. when satellite mission is extended using inclined orbit satellite) are not considered here.

NOTE 3: Uplink frequency: 30,0 GHz

NOTE 4: Downlink frequency: 20,2 GHz

6.2.2 Frequency accuracy

The frequency accuracy of the terminal burst is the result of a number of contributors, some of which are independent of the terminal speed (i.e. of the terminal-related Doppler Effect). Typical fixed contribution - i.e. the frequency accuracy typical of a classical system - is provided in clause 10 (Table 10.2).

The frequency accuracy provided in Table 10.2 is typical of that obtained in a DVB-RCS(2) satellite system. In the case of a mobile environment, the major additional contribution is due to the terminal motion. As explained before, it induces two types of effects: the first one is related to the induced Doppler on the NCR received reference (which derives in a frequency offset on the terminal transmit frequency). The second one is a frequency offset on the return uplink path (from terminal to satellite).

The resulting frequency Doppler shift provided in Tables 6.3 and 6.4 includes these two contributions (one relevant to downlink Doppler and the other to uplink Doppler), but both applying to the uplink frequency.

The tolerable burst frequency offset within the gateway modem is dependent on the gateway receiver modem implementation, and as is such not directly specified in the DVB-RCS2 standard. A representative set of values for acceptable burst frequency offset that range from 0,5 % to 3 % of a symbol rate is considered instead, in agreement with DVB-RCS2 system and gateway manufacturers.

This allows to derive several combinations of maximum terminal speed and compatible terminal symbol rates. The following analysis assumes that the terminal frequency burst accuracy is the same at initial access of the terminal (logon burst) as for the subsequent traffic bursts. This relies on the assumption that no specific enhancement on the logon burst is performed to facilitate its demodulation and frequency detection. It also means that the traffic burst does not further benefit from frequency correction provided by the network. The discussion is hereafter provided on that assumption, covered by the current DVB-RCS2 standard and allowing to extend the range of standard applicability to mobile services.

The analysis is performed both for Ku-band and Ka-band.

Table 6.3 summarizes for the Ku-band case the minimum symbol rate requirement in order to be compatible with the aggregated frequency shift generated by the terminal motion and the fixed contribution detailed in Table 10.2.

Table 6.3: Minimum symbol rate requirement as a function of terminal speed (Ku-band)

	Pedestrian	Maritime	Vehicular (bus car, truck)	Train	Aeronautical (< speed of sound)	Fixed Terminal
Speed of the terminal	5 km/h	25 km/h	120 km/h	350 km/h	1 188 km/h	0 km/h
Freq. Doppler Shift (U/L and D/L)	134 Hz	671 Hz	3 222 Hz	9 398 Hz	31 900 Hz	0 Hz
Fixed contribution	1 590 Hz	1 590 Hz	1 590 Hz	1 590 Hz	1 590 Hz	1 590 Hz
Aggregated Frequency Drift	1 724 Hz	2 261 Hz	4 812 Hz	10 988 Hz	33 490 Hz	1 590 Hz
Symbol rate frequency accuracy	**Minimum symbol rate (ksym/s)**					
0,5 %	345	452	962	2 198	6 698	318
1 %	172	226	481	1 099	3 349	159
2 %	86	113	241	549	1 675	80
3 %	57	75	160	366	1 116	53

Figures 6.1 and 6.2 present the allowed terminal speed as a function of the symbol rate of the terminal for the different acceptable burst frequency accuracy within the gateway modem (expressed as a percentage of the symbol rate). The range of symbol rate is intentionally limited to about 2 Msym/s, in order to remain in representative target rates in terms of service (typically from a few kbits/s to 1 Mbits/s).

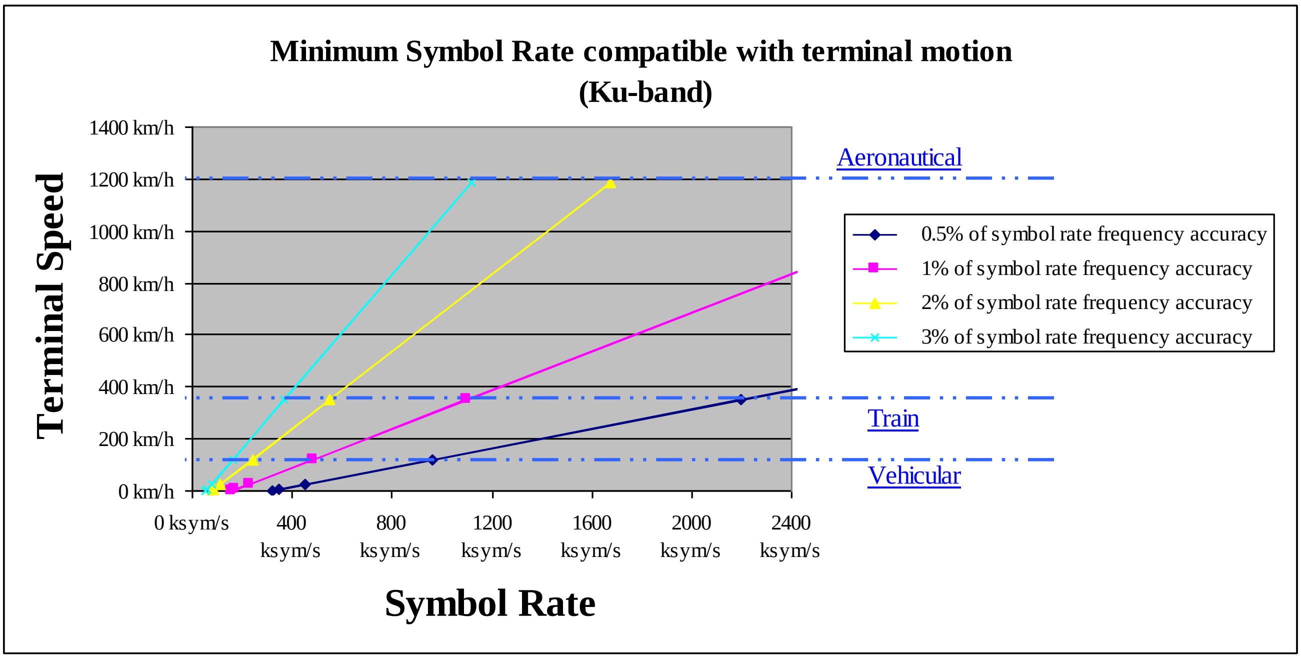


Figure 6.1: Minimum symbol rate compatible with high-speed terminal motion (Ku-band)

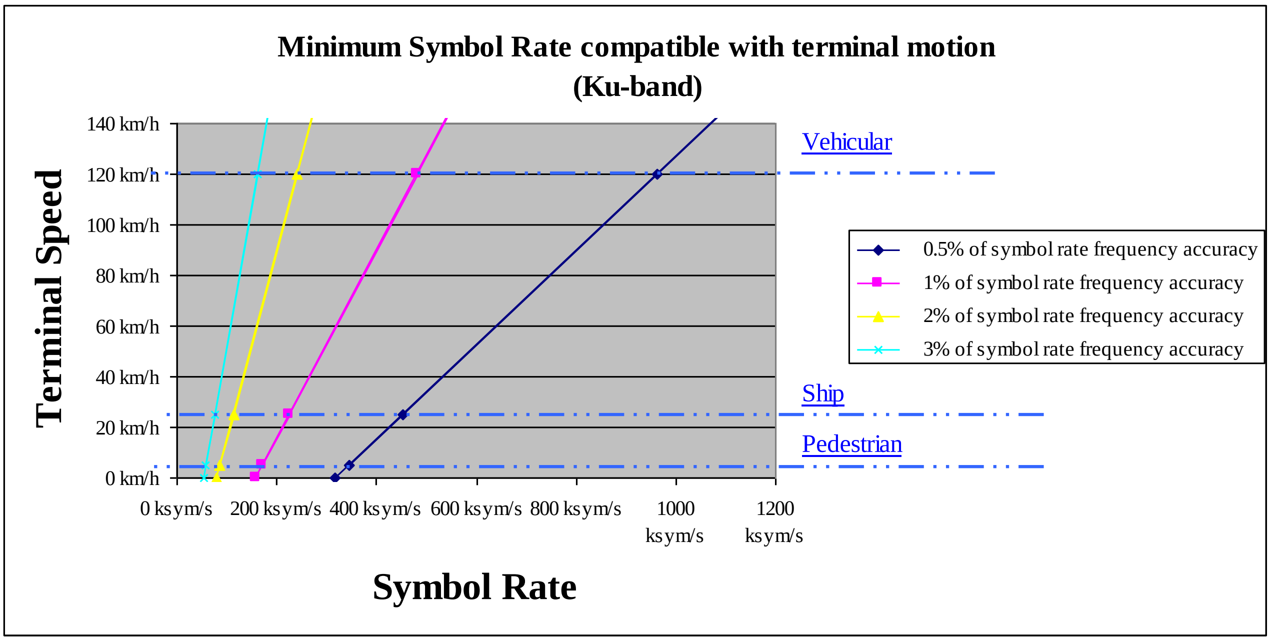


Figure 6.2: Minimum symbol rate compatible with low-speed terminal motion (Ku-band)

Table 6.4 summarizes for the Ka-band case the minimum symbol rate requirement in order to be compatible with the aggregated frequency shift generated by the terminal motion and the fixed contribution detailed in Table 10.2.

Table 6.4: Minimum symbol rate requirement as a function of terminal speed (Ka-band)

	Pedestrian	**Maritime**	**Vehicular (bus car, truck)**	**Train**	**Aeronautical (< speed of sound)**	**Fixed Terminal**
Speed of the terminal	5 km/h	25 km/h	120 km/h	350 km/h	1 188 km/h	0 km/h
Freq. Doppler Shift (U/L and D/L)	278 Hz	1 389 Hz	6 667 Hz	19 444 Hz	66 000 Hz	0 Hz
Fixed contribution	3 784 Hz	3 784 Hz	3 784 Hz	3 784 Hz	3 784 Hz	3 784 Hz
Aggregated Frequency Drift	4 062 Hz	5 173 Hz	10 451 Hz	23 228 Hz	69 784 Hz	3 784 Hz
Symbol rate frequency accuracy	**Minimum symbol rate (ksym/s)**					
0,5 %	812	1 035	2 090	4 646	13 957	757
1 %	406	517	1 045	2 323	6 978	378
2 %	203	259	523	1 161	3 489	189
3 %	135	172	348	774	2 326	126

Figures 6.3 and 6.4 represent the range of minimum symbol rates for maximum allowable terminal speed. The range of symbol rate is here again intentionally limited to about 2 Msym/s, in order to remain in representative target rates in terms of service (typically from a few kbits/s to 1 Mbits/s).

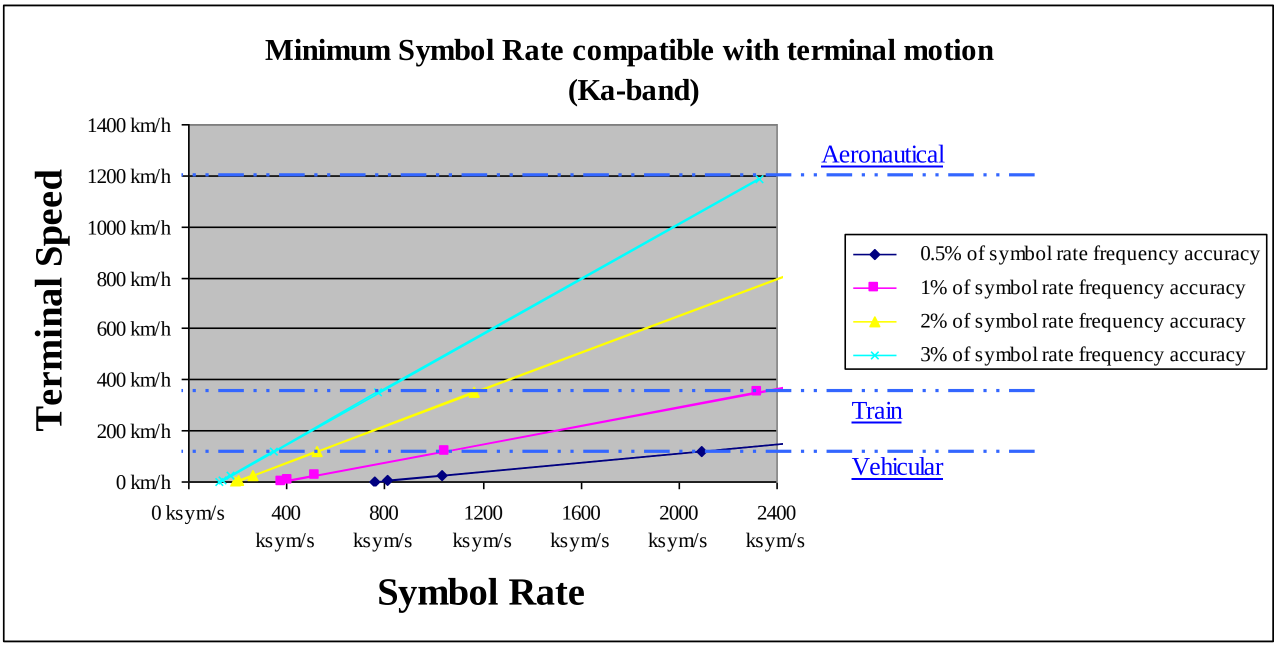


Figure 6.3: Minimum symbol rate compatible with high-speed terminal motion (Ka-band)

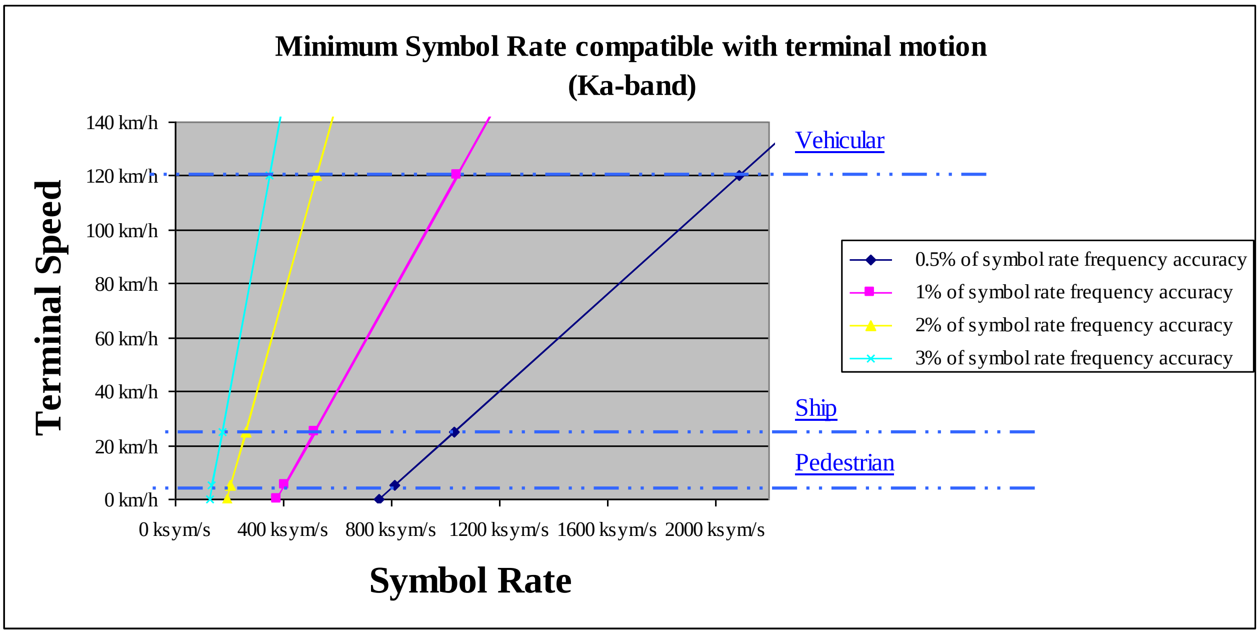


Figure 6.4: Minimum symbol rate compatible with low-speed terminal motion (Ka-band)

Figures 6.1 to 6.4 give some combinations of terminal speed and transmit symbol rates which are feasible within the DVB-RCS2 standard definition, for the defined typical gateway performance.

The values obtained rely on the assumption that the maximum defined acceptable frequency offset is applicable for both initial burst (logon) and traffic and control bursts, assuming that no frequency correction is performed. These values could be reduced (and the range of applicability improved) in case tolerance for logon burst frequency offset is improved and frequency correction performed. In particular, the minimum rates for aeronautical applications could be reduced to better reflect application needs (512 kbits/s, 1 024 kbits/s).

NOTE: When the speed and the targeted symbol rate of the mobile terminal are not within the defined envelope, operation may be facilitated by introducing some frequency Doppler pre-compensation mechanisms within the terminal (e.g. by using GPS location and by speed and direction information about the vehicle - aircraft for example-, or through frequency offset estimation deduced from forward downlink reception). The pre-compensation, which is not in the current standard definition, would allow operating conditions similar to those obtained in the non-mobile environment.

6.2.2.1 Frequency and timing drift within the burst

The frequency drift and timing drifts that will occur within the burst impact the burst demodulation performances, and may constrain the burst duration, thus the applicable burst formats and the profiles for some applications.

The frequency drift is mainly due to the terminal acceleration. Assuming that the frequency is estimated on the preamble, frequency drift will induce phase rotation within the burst, with a maximum value on the last symbol of the burst. Assuming a typical value of 4° maximum phase rotation for acceptable degradation, a maximum burst duration for each mobile applications can be defined.

NOTE: The above paragraph is a worst case assumption. Implementations exist where frequency detection is made on the whole burst or where phase tracking can be made over the burst. In that case, the phase rotation on any symbol within the burst can be relaxed significantly.

Tables 6.5 and 6.6 provide the worst case maximum burst duration values for both Ku-band and Ka-band, considering the frequency drifts defined in Tables 6.1 and 6.2.

Table 6.5: Example of maximum burst duration for acceptable impact of frequency drift on the return link (Ku-band)

Type of mobile	Uplink Frequency Drift (Hz/s)	Maximum burst duration (ms)
Pedestrian	48	21,4
Maritime	242	9,6
Vehicular (bus, car, truck)	483	6,8
Train	242	9,6
Aeronautical (< speed of sound)	822	5,2
NOTE: The condition for acceptable impact of frequency drift on a burst is a 4° maximum phase drift.		

Table 6.6: Example of maximum burst duration for acceptable impact of frequency drift on the return link (Ka-band)

Type of mobile	Uplink Frequency Drift (Hz/s)	Maximum burst duration (ms)
Pedestrian	100	14,9
Maritime	500	6,7
Vehicular (bus, car, truck)	1 000	4,7
Train	500	6,7
Aeronautical (< speed of sound)	1 700	3,6
NOTE: The condition for acceptable impact of frequency drift on a burst is a 4° maximum phase drift.		

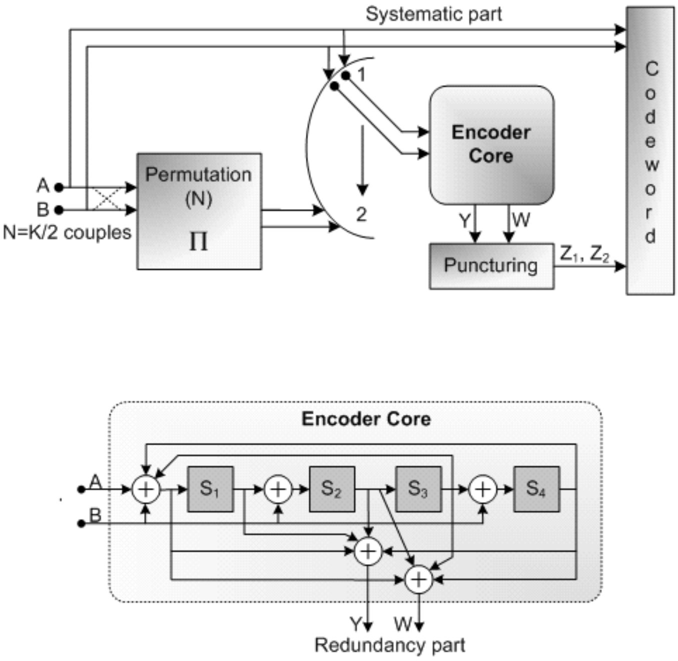
Adequate burst formats should be selected in order to remain within the above constraints.

Concerning time drift, it is assumed that the timing drift resulting from both symbol timing inaccuracy and Doppler Effect should not induce a timing error of any symbol within the burst higher than of 0,1 symbol duration.

6.3 Physical Layer

6.3.1 Turbo-Phi Encoder

The Turbo-Phi encoder architecture used for linear modulation in the normative document [i.1] is shown in Figure 6.5. It is based on a parallel concatenation of two double-binary Recursive Systematic Convolutional (RSC) encoders, fed by blocks of *K* bits (*N* = *K*/2 couples). For this encoder, *N* should not be a multiple of the linear shift register period (i.e. 15 for the 16-state code). *N* should be a multiple of 4, because of the permutation law Π.



NOTE: Redundancy y_2 is only used for coding rates less than 1/2.

Figure 6.5: Structure of the proposed 16-state double-binary turbo encoder

The encoding of a block involves encoding the information sequence in the natural order (switch in position "1"), permuting the data (interleaver Π) and encoding the information sequence in the natural order (switch in position "2"). At the end of the encoding process, the final state is the same as the initial state. Such a code can be represented using a circular trellis. For each block of user bits each component encoder computes a state, denoted the circular state, in such a way that the trellis "tail-bites" itself, making the trellis circular: by initializing the encoder's shift register with this circular state, the shift register terminates in the same state when all user bits have been fed into the encoder.

The permutations (i.e. interleavers) between the component convolutional codes is based on simple algebraic laws, avoiding the use of memory-consuming look-up tables for the permutations. The laws are independent of the code rates and have been fine-tuned for each block size to avoid flattening of the error curve for BER above 10^{-9}.

6.3.2 Assessing the SISO decoder complexity

Figure 6.6 gives the generic processing engine of an associated turbo decoder. This engine is built around two soft-in/soft-out modules (SISO). The SISO are identical in structure, however, as inputs, one receives data in the natural order and the other one in the interleaved order. The outputs of one SISO, after proper scaling and after reordering, are used by its dual SISO in the next step.

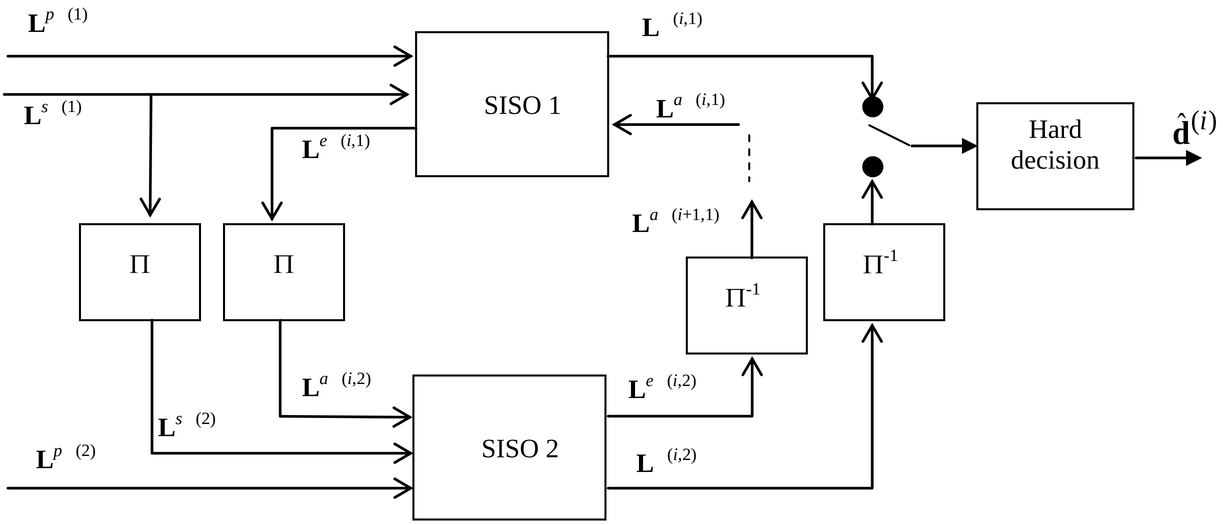


Figure 6.6: The principle of the turbo decoding

6.3.2.1 Max-Log-MAP decoding of a RSC code

In practical hardware turbo decoders, the so-called Max-Log-MAP algorithm [i.48] is adopted to implement the SISO decoders. The Max-Log-MAP algorithm is derived from a simplified version of the symbol-by-symbol *Maximum A Posteriori* (MAP) algorithm, also known as the Bahl, Cock, Jelinek and Raviv (BCJR) algorithm [i.41] in the logarithmic domain. The BCJR algorithm uses the trellis of the code to computes, for each data symbol, an *A Posteriori Probability* (APP) by evaluating the probabilities of all possible paths from the initial state to the final state in the trellis.

Metrics computed in practice in this algorithm are proportional to the probabilities computed in the original MAP algorithm:

$$met = -\sigma^2 \log p \tag{1}$$

Moreover, the so-called Max-Log approximation is adopted:

$$\ln(\exp(a) + \exp(b)) \approx \max(a,b) \tag{2}$$

the max operator becoming a min one in practice due to the minus sign in (1).

Let us consider a Recursive Systematic Convolutional (RSC) code with the following parameters:

- ν is the memory length of the code,
- *m* is the size of the input symbols at the encoder input (the code is said to be *m*-binary),
- *n* is the number of coded bits provided by the encoder at each trellis stage, when no puncturing is performed.

We will usually represent the k^{th} *m*-binary information symbol $\mathbf{d}_k = (d_{k,1} \cdots d_{k,m})$ at the encoder input by the scalar quantity $\delta_k = \sum_{j=1}^{m} 2^{j-1} d_{k,j}$, taking values between 0 and $2^m - 1$, and we write $\mathbf{d}_k \equiv \delta_k$. The whole information sequence $(\mathbf{d}_0 \cdots \mathbf{d}_{N-1})$ is denoted by **d**. We denote by $\mathbf{u} = (\mathbf{u}_0 \cdots \mathbf{u}_{N-1})$ the sequence of corresponding encoded and modulated symbols ($\mathbf{u}_k = (u_{k,1} \cdots u_{k,n})$ with $u_{k,l} = \pm 1$) and by $\mathbf{v} = (\mathbf{v}_0 \cdots \mathbf{v}_{N-1})$ the sequence observed at the channel output or equivalently at the decoder input ($\mathbf{v}_k = (v_{k,1} \cdots v_{k,n})$). Here, sequence **v** is a concatenation of the sequences $\mathbf{L}^s$ and $\mathbf{L}^p$. Symbols $u_{k,l}$ and $v_{k,l}$ correspond to systematic bits for $l \leq m$ and to parity bits for $l > m$.

The Max-Log-MAP algorithm purpose is the computation of the *a posteriori* log-likelihoods $L_k(\delta)$ defined as:

$$L_k(\delta) = -\frac{\sigma^2}{2} \log \Pr(\mathbf{d}_k \equiv \delta | \mathbf{v}) \quad , \ \delta = 0, \cdots, 2^m - 1 \tag{3}$$

The decoding algorithm involves:

- **The computation of branch metrics** $c_k(s',s)$ at each time step k is written as:

$$c_k(s',s) = 2L_k^a(\delta) - \sum_{l=1}^{m} v_{k,l}.u_{k,l} + c_k^e(s',s) \qquad (4)$$

- **The computation of forward and backward state metrics** for each trellis state s at each time step k:

The state metrics are computed recursively using the following relations:

Forward recursion:

$$a_k(s) = \min_{s'}(a_{k-1}(s') + c_{k-1}(s',s)) \text{ for } k = 1,\cdots,N \qquad (5)$$

Backward recursion:

$$b_k(s) = \min_{s'}(b_{k+1}(s') + c_k(s,s')) \text{ for } k = N-1,\cdots,0 \qquad (6)$$

- **The computation of the a posteriori log-likelihood** $L_k(\delta)$ at each time step k:

If we denote by $\lambda_k(\delta)$ the soft information defined as:

$$\lambda_k(\delta) = \min_{(s',s)}\left(a_k(s') + c_k(s',s) + b_{k+1}(s)\right) \qquad (7)$$

the *a posteriori* log-likelihood related to data vector $\mathbf{d}_k$ is computed as:

$$L_k(\delta) = \frac{1}{2}\left(\lambda_k(\delta) - \min_{\delta'} \lambda_k(\delta')\right) \qquad (8)$$

- **The computation of the hard decision** at each time step k:
    - Actually, the hard decision provided by the decoder correspond to the binary representation of δ that minimizes $\lambda_k(\delta)$ and makes $L_k(\delta)$ equal to zero.

$$\hat{\delta} = \arg\min_{\delta}\left(L_k(\delta)\right) = \arg\min_{\delta}\left(\lambda_k(\delta)\right) \qquad (9)$$

- **The computation of the extrinsic information** $L_k^e(\delta)$ at each time step k:
    - The extrinsic information computation is similar to the computation of the *a posteriori* log-likelihood $L_k(\delta)$**,** using the extrinsic branch metrics. We compute the extrinsic soft information $\lambda_k^e(\delta)$ defined as

$$\lambda_k^e(\delta) = \min_{(s',s)}\left(a_k(s') + c_k^e(s',s) + b_{k+1}(s)\right) \qquad (10)$$

- and then the extrinsic log-likelihood $L_k^e(\delta)$ as follows

$$L_k(\delta) = \frac{1}{2}\left(\lambda_k^e(\delta) - \lambda_k^e(\hat{\delta})\right) \qquad (11)$$

- The term subtracted to $\lambda_k^e(\delta)$ is the extrinsic value corresponding to the hard decision $\hat{\delta}$.

Due to the application of the Max-Log approximation (2), there is a systematic overestimation of all metrics and this algorithm turns to be sub-optimal. In order to cope with this problem, a scaling operation of the extrinsic information is performed in the turbo decoder: we typically use value 0.7 except for the last iteration where extrinsic information is not scaled. Note that with the definitions of the metrics given in (1), the Max-Log-MAP algorithm does not need the estimation of the noise variance σ^2 .

6.3.2.2 Computational complexity of the Max-Log-MAP algorithm

In this clause, we make a list of basic operations involved in the algorithm, such as additions, comparisons, etc.

Branch metrics computation

The trellis of an *m*-binary RSC code with memory ν is composed of 2^ν with 2^m transitions starting from and arriving to each state. At each step *k*, there are 2^n different branch metrics to compute, corresponding to the 2^n possible values of each vector $\mathbf{u}_k$. In (4), term $\sum_{l=1}^{n} v_{k,l}.u_{k,l}$ can be actually written as $\sum_{l=1}^{n} \pm v_{k,l}$. The computation of all combinations of $\pm v_{k,l_1} \pm v_{k,l_2}$ requires 4 additions: $v_{k,l_1} + v_{k,l_2}$, $v_{k,l_1} - v_{k,l_2}$, $-\left(v_{k,l_1} + v_{k,l_2}\right)$ and $-\left(v_{k,l_1} - v_{k,l_2}\right)$ (the inversion of a number is similar to an addition). Hence, all branch metrics of (5) can be computed with an $(n-1)$-stage addition tree, resulting in a total of $4\left(2^{n-1}-1\right)$ additions. The addition of the *a priori* term requires 2^m extra additions and 2^m multiplications by 2.

The computation of the branch metrics can be performed twice, once for the forward recursion and once for the backward recursion or can be performed once and the metrics have then to be stored.

State metrics computation

The update of one forward state metric according to (5) involves the comparison and selection of 2^m concurrent paths that can be implemented using 2^m additions and $2^m - 1$ comparison-selection operations, implementing a tree structure. The update of backward state metrics requires the same number of operations. Since the trellis has 2^ν states, one recursion step (forward or backward) requires $2^{\nu+m}$ additions and $2^\nu\left(2^m-1\right)$ comparison-selection operations.

***A posteriori* log-likelihood and hard decision computation**

The computation of the *a posteriori* log-likelihoods according to (7) and (8) requires the computation of the 2^m values of $\lambda_k(\delta)$, $\delta = 0,\cdots,2^m-1$. Relation (10) involves two additions for each transition in the trellis. This complexity can be reduced to one addition by observing that partial terms $a_k(s') + c_k(s',s)$ or $c_k(s',s) + b_{k+1}(s)$ are already available through the forward or backward recursion. For each value of δ, the minimum value of the 2^ν terms $a_k(s') + c_k(s',s) + b_{k+1}(s)$ has to be found, resulting in $2^\nu - 1$ compare-select operations, using a tree structure.

Consequently, the computation of the 2^m values for $\lambda_k(\delta)$ requires $2^{m+\nu}$ additions and $2^m\left(2^\nu-1\right)$ comparisons and selections.

The computation of the 2^m *a posteriori* log-likelihoods requires a compare and select tree to compute the min term in (11), that is 2^m-1 compare and select operations, 2^m subtractions and 2^m divisions by 2. Actually, the subtraction in the case of $\lambda_k(\hat{\delta})$ can be avoided, since $L_k(\hat{\delta}) = 0$ and the number of subtractions can be reduced to 2^m-1. The hard decision $\hat{\delta}$ can be inferred from the compare and select tree allowing the minimum value of $\lambda_k(\delta)$ to be computed.

Extrinsic information computation

For each symbol δ, the extrinsic information is computed with (10). If we assume that terms $L_k^a(\delta) - \frac{1}{2}\sum_{l=1}^{m} v_{k,l}.u_{k,l/\mathbf{d}_k\equiv\delta}$ have already been made available during the branch metric computation step using the decomposition proposed in (7), each piece of extrinsic information is obtained with one addition and one subtraction. The total extrinsic information computation is then performed using 2^m additions and 2^m subtractions. Note that, in practice, the decoder does not need to provide the complete *a posteriori* log-likelihoods, since only extrinsic pieces of information are exchanged between the component decoders and only the hardware decision needs to be known at the end of the iterative process.

6.3.2.3 Summary of Turbo Decoder complexity

Table 6.7 summarizes the resulting complexity for the process of a trellis stage, or equivalently of an information symbol. The complexity of the multiplications or divisions by the factor 2 can be neglected, since they can be implemented by a simple shift to the left or to the right.

Adding the respective complexities of the computation of the branch metrics, the forward and backward recursions, the soft and binary outputs, and the extrinsic log-likelihoods, leads to a complexity of $2^m\left(2^{\nu+1}+4\right)+8\left(2^{n-1}-1\right)-1$ additions and $2^{\nu+m+1}+2^m-2^\nu-2$ compare-select operations.

For the DVB-RCS code, the code parameters are $\nu = 3$ (8-state code), $m = 2$ (double-binary code), $n = 4$ (RSC rate ≥ 1/3). The computational complexity amounts to 171 additions and 79 compare-select operations.

For the DVB-RCS2 code, the code parameters are ($\nu = 4$, $m = 2$, $n = 4$), 267 additions and 159 compare-select operations are required.

Table 6.7: Computational complexity of the Max-Log-MAP algorithm

	Add (or subtract)	Compare-select	Mul or div by 2
Branch metrics (forward or backward)	$4\left(2^{n-1}-1\right)+2^m$		2^m
One step of recursion (forward or backward)	$2^{\nu+m}$	$2^\nu\left(2^m-1\right)$	
Computation of $\lambda_k(\delta)$	$2^{\nu+m}$	$2^m\left(2^\nu-1\right)$	
A posteriori Log-likelihoods and hard decision	2^m-1	2^m-1	2^m
Extrinsic Log-likelihoods	2^{m+1}		
Total computational requirement per information symbol (or for *m* information bits)	$3\times 2^{\nu+m}+5\times 2^m$ $+8\left(2^{n-1}-1\right)-1$	$2^\nu\left(3\times 2^m-2\right)-1$	2^{m+1}

6.3.3 CPM complexity

This clause provides an overview of algorithmic complexity of implementing CPM scheme at the transmitter and the receiver.

6.3.3.1 CC-CPM Modulator

Two binary, non-systematic, non-recursive convolutional codes have been selected. One can easily shift from 5/7 to 15/17 encoders as shown in Figure 6.7. The multiplexer can be controlled with the `PrecoderFlag signal`.

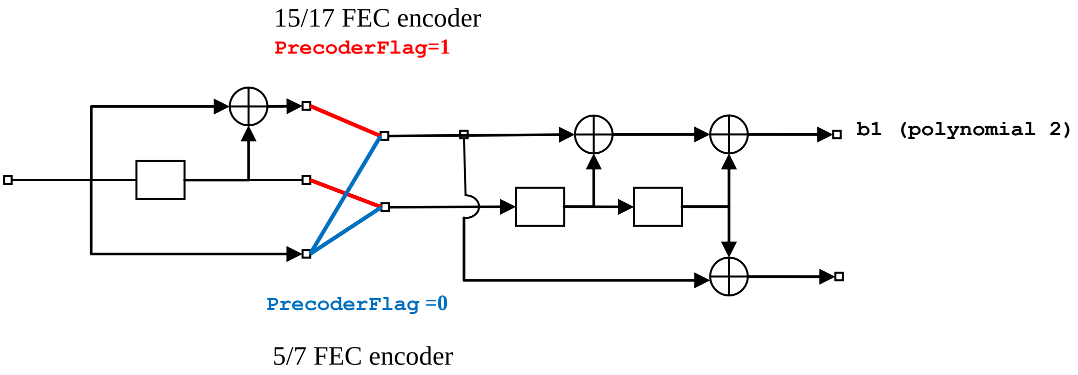


Figure 6.7: CC encoder structure

6.3.3.2 CPM receiver

Figure 6.8 shows the architectural diagram of a simplified receiver.

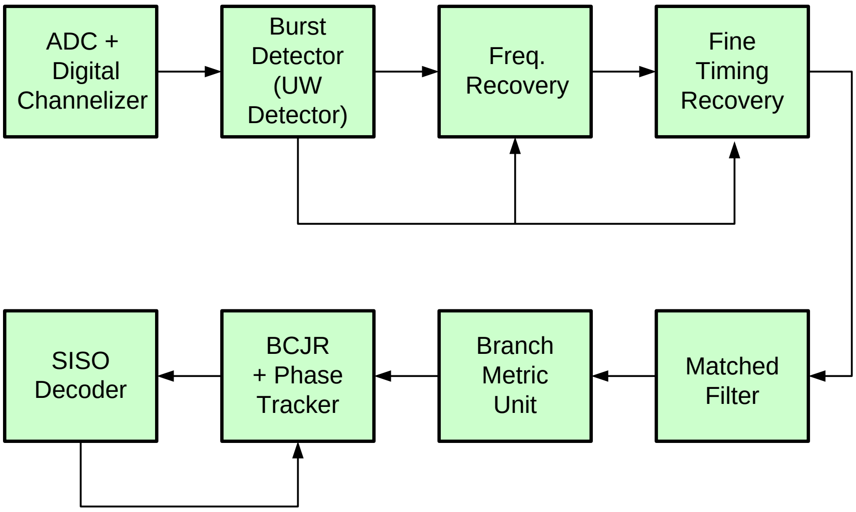


Figure 6.8: CPM receiver architecture

Channelization Block

This channelization block represents digital down conversion from low IF (Intermediate Frequency) band and separate the desired signal from adjacent signal through Low Pass Filtering (LPF). The LPF is a significant factor of the system because it influences the performance when adjacent carrier interference is considered. In particular, the passband and stop frequencies have been selected according to the Power Spectral Density (PSD) of the signal, which depends on the modulation parameters such as CPM encoder memory length (L), Alphabet Size (M) and Modulation index (h). When the configurations with high spectral efficiencies are considered, the cut off frequency of the filter should be chosen in order to find a good trade-off between the amount of interference coming from the adjacent channels and the amount of signal power of the useful carrier after the filtering.

Synchronization Block

The synchronization block is composed with UW (Unique Word) detection, carrier frequency offset recovery and fine timing recovery. By using two distributed pilot symbol like preamble and midamble, it enables to obtain good performance through special correlator. It can be designed to estimate the timing offset and carrier frequency offset through FFT (Fast Fourier Transform) computation, jointly. Among various possible approaches, the use of data-aided (DA) frequency estimator based on the Rife and Boorstyn (R&B) algorithm [i.42] can be considered. The received signal in AWGN channel can be expressed as:

$$r(t) = s(t,\alpha)e^{j[2\pi\nu t+\theta(t)]} + n(t)$$

where ν is the unknown frequency offset, and θ(t) corresponds to the carrier phase-noise. The frequency estimation technique consists of the following steps:

a) Correlation functions

Over the two known fields (i.e. the preamble and the midamble), the following correlation sequences are calculated:

$$z_n^{pre} = \int_{nT}^{nT+T} r(t)s^*(t,\alpha)dt \qquad n = 0,1,...N_{pre} - 1$$

$$z_n^{mid} = \int_{(n+N_{pre}+D)T}^{(n+N_{pre}+D)T+T} r(t)s^*(t,\alpha)dt \qquad n = 0,1,...N_{mid} - 1$$

where the parameter *D* takes into account for N_{dist} and the number of "normalization sequence" symbols.

b) FFT computation

The FFT is applied on both sequences as follows:

$$Z_k^{pre} = \sum_{n=0}^{\rho(N_{pre}+D+N_{mid})-1} z_n^{pre} e^{-j2\pi \frac{kn}{\rho(N_{pre}+D+N_{mid})}}$$

$$Z_k^{mid} = \sum_{n=0}^{\rho(N_{pre}+D+N_{mid})-1} z_n^{mid} e^{-j2\pi \frac{kn}{\rho(N_{pre}+D+N_{mid})}}$$

where ρ is the R&B pruning factor. The incoming sequences are zero-padded, so that they have the same length, $\rho(N_{\mathrm{pre}}+D+N_{\mathrm{mid}})$.

c) Sequence combination

The two frequency-domain sequences are combined in order to obtain the following decision vector:

$$Z_k = Z_k^{pre} + Z_k^{mid} e^{-j2\pi \frac{k(N_{pre}+D)}{\rho(N_{pre}+D+N_{mid})}} \qquad k = 0,1,\ldots(N_{pre}+D+N_{mid})-1$$

d) Search for the maximum value

Finally, the value $\hat{k}$ of k corresponding to the maximum value of $|Z_k|^2$ is selected. Then, the carrier frequency estimate is computed as:

$$\hat{\nu} = \frac{\hat{k}}{\rho(N_{pre}+D+N_{mid})T}$$

Matched Filter Block

The most convenient approach for deriving low complexity algorithms for CPM digital demodulation consists of resorting to proper approximations of the original CPM waveform. In particular, the adopted technique is based on the Laurent decomposition [i.52]. The complex envelope of a CPM signal could be expressed as:

$$s(t,\alpha) = \sum_{k=0}^{F-1} \sum_n \beta_{k,n} p_k(t-nT)$$

where $F=(M-1)*2^{(L-1)\log M}$ is the number of linearly modulated pulses $p_k(t)$, and $\beta_{k,n}$ are the so-called pseudo symbols. The exact expressions of pulses $p_k(t)$ and those of symbols $\beta_{k,n}$ as a function of the CPM parameters and of the information symbols can be found in [i.52]. For reducing the demodulation complexity, only $S<F$ conditional terms are considered in the CPM front-end linear filtering. The first M-1 modulated pulses are called *principal components*, and should be sufficient to collect almost all the transmitted energy for L<3. It is assumed $S=(M-1)M^{L-1}$.

The Laruent filters can be implemented as Finite Impulse Response (FIR) filters. The complexity is determined by the number of selected filter among the number of complete filters. When the impulse responses of two components are very close to one another, the two components can be implemented as a single filter.

Computation of branch metrics and BCJR algorithm

The branch metrics computation is a demanding task in the CPM receiver.

It requires $Nc\,(the\ number\ of\ selected\ \ filter)$ $Nc\ (the\ number\ of\ selected\ filter)$complex multiplications per branch over CPM detection trellis with the forward and backward computations for each branch. One trellis section has $M\ (Alphbet\ size) \times p$ $M(Alphbet\ size) \times p$(Modulation index) branches if only principal components are selected, and $M^{L(Memory)} \times p$ if more components are considered. Overall, it requires $M^{L(Memory)} \times p \times Nc$ complex multiplication per trellis section during one iteration. One trellis section has $M \times p$ $M \times p$branches when principal components only are selected and $M^2 \times p$ if more components are taken into account.

Phase Tracking

In the presence of phase-noise, the phase synchronization can be embedded in the BCJR algorithm [i.41] by using the Bayesian approach, and consists of assuming a probabilistic model for the phase noise process. In practice, the algorithm is obtained by discretizing the channel phase process in only R possible phase values:

$$\{2\pi i / R\}_{i=o}^{R-1} .$$

The proposed value of R depends on the value of *p*, i.e. the denominator of the CPM modulation index *h*. The recommended values of R (as determined based on software simulation) are reported below.

p	R
5	6*p*
4	8*p*
3	8*p*
2	12*p*
Any other value of *p*	4*p*

More details on this algorithm and the derivation of the discretized phase estimation technique can be found in [i.53]. However, the discretized phase algorithm is computationally demanding. To reduce the complexity, the simplified algorithm can be considered, as described in [i.54]. However, the complexity reduction will cause performance degradation (1 dB performance degradation at PER = 10^{-3} has been reported based on software simulations).

6.3.4 Trellis Termination in CPM

Trellis termination is applied at the CPM modulator in order to force the modulator to a known state prior to inserting the unique word (UW) symbols. As an example, Figure 6.9 shows the location of the trellis termination symbols within a burst of symbols.

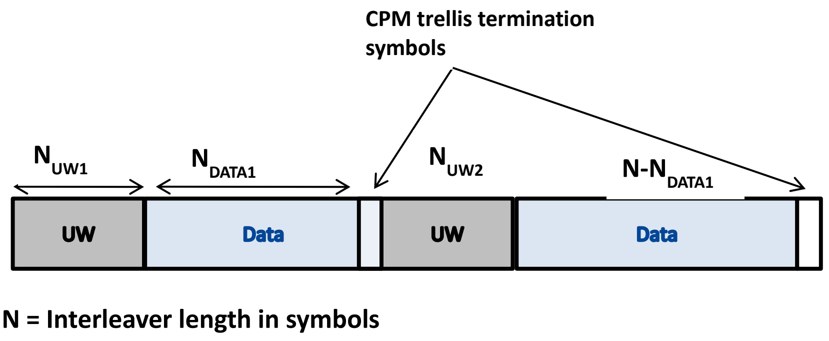

CPM trellis termination symbols
N_UW1
N_DATA1
N_UW2
N-N_DATA1
UW
Data
UW
Data
N = Interleaver length in symbols

Figure 6.9: CPM burst structure

Trellis termination ensures that the CPM waveforms generated using the UW symbols are data invariant, due to which they can be conveniently inferred/ reproduced at the receiver.

The CPM signal phase can be completely specified using a continuous phase encoder(CPE) followed by the memoryless modulator (MM) as shown in Figure 6.10.

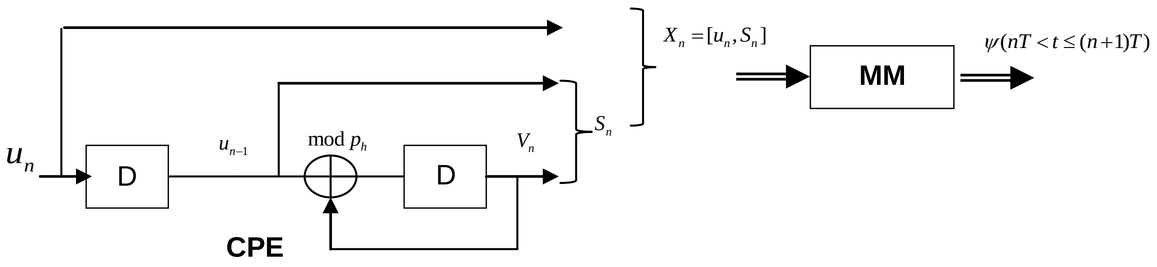


Figure 6.10: CPM Modulator as a continuous phase encoder followed by a memoryless modulator

It should be noted that:

- The CPE is a linear-time invariant sequential circuit.
- The memory in the modulation is captured in the CPE which the MM uses to generate the CPM phase.
- The number of states is $p_h \times M$. Also, the finite-state machine is time-invariant.
- The phase trajectories in any two symbol intervals are time translates of one another [i.39].

Trellis termination involves driving the CPE to a known state, typically the all-zero state, the additional symbols required to do so are called the "tail-symbols". Due to its recursive nature, it is not possible to terminate the CPM trellis by transmitting L '0' tail-symbols. The tail-symbols will depend on the state of the encoder, after, the *N* data symbols are encoded by the CPE. Since the CPM state depends on the *M*, *L* and p_h, the tail symbols will also depend on the choice of these parameters. Clause 7.3.7.2.3 of the normative document [i.1], specifies the tail-symbols required for different selections of *M*, *L* and p_h,.

6.4 Return Link Encapsulation

6.4.1 RLE Principles

The return link encapsulation (RLE) is designed to convey higher layer packets, such as IP datagrams, Ethernet frames, MPEG units or signalling packets, into a return link burst. As a design feature, the RLE protocol adopted for DVB-RCS2, is a modification of GSE (used on the forward link) but with lower overhead and specially tailored to the return-link properties. As such its implementation complexity is comparable if not slightly less than the overhead and complexity of GSE. Clause 6.4.1 and its subclauses describe the position of the protocol in the overall system and describe the protocol.

Figure 6.11 illustrates an encapsulation example for the return link. IP datagrams (or other network layer protocol units) are fragmented according to the payload size of the bursts allocated by the resource management and according to scheduling decisions which may be driven by QoS requirements. Each fragment is put into an RLE packet which starts with an RLE header followed by data and one or more of the resulting RLE packets are packed into a return link burst and transmitted.

Figure 6.11 shows several features of the protocol. The first, long, IP packet is followed by two short ones. It is assumed that the long one has lower priority and the second and third a higher. As can be seen the fragmentation of the first one can be suspended and the second and third will be sent before the first packet is resumed. Also the physical layer bursts have different sizes (in terms of payload bits but not symbols), because the module and coding changed between the first burst and the second one.

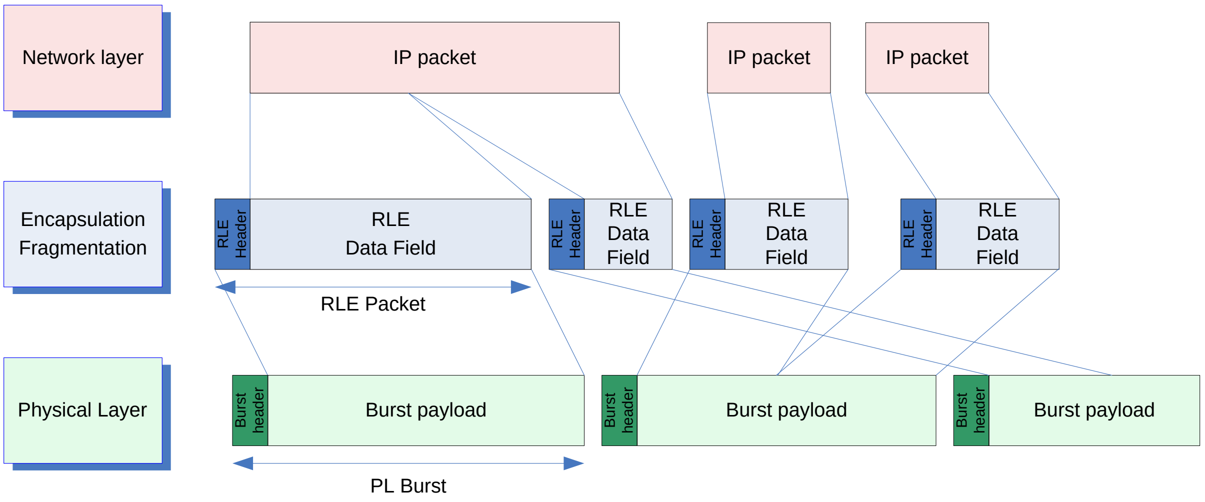


Figure 6.11: Return Link Encapsulation Example

6.4.2 RLE Interfaces

This clause provides informative material on conceptual RLE transmitter interfaces. The interfaces are conceptual because the description does not provide an implementation guideline. Rather logical interfaces that correspond to information flows between RLE and other modules of the terminal are described. These interfaces may or may not map to actual, physical interfaces in an implementation.

6.4.2.1 RLE transmitter external interfaces

The RLE transmitter is located in the terminal. Its conceptual interfaces are shown in Figure 6.12. It is assumed that RLE is integrated with the scheduler and the scheduler queues for optimum performance. The functions of the interfaces are the following:

- **L2-IH:** This is the input interface from the higher layers. RLE receives higher layer packets (mainly IP packets) with some attached information:
    - The protocol type. This will be IPv4 (0x0800) or IPv6 (0x86dd), but other types are also possible (see Table 5-1 in normative document [i.1]).
    - The packet label. This is a one-byte value containing the upper byte of the MAC24 to send the packet from.
    - A list of extension headers. This list may be empty.
    - The higher layer packet. For some types of extension headers this may be empty.
    - A QoS tag. This optional tag is used by the scheduler and may contain traffic classes, priorities. The RLE module encapsulates the packet and puts the result into one of the scheduler queues.
- **L2-IS:** This is the interface for L2 M&C signalling packets that have to be sent to the hub. The protocol type of these packets is fixed to 0x0082, there may be no extension headers and no label. The RLE module puts these packets into one of its scheduling queues.
- **L2-MQ:** This is the interface for an optional congestion control module. The module may monitor the queue fill states, fill rates and drain rates, derive congestion signals and trigger packet dropping or traffic shaping in the queues.
- **L2-MR:** This is the interface to the request manager. The request manager monitors the fill states, fill rates and drain rates of the scheduler queues and produces resource requests to be sent to the hub.
- **L2-MB:** On this interface the RLE module receives burst descriptions for the bursts it needs to produce. These descriptions contain the burst size for use by the burst packing sub-layer and may contain additional information to be passed to the PHY (timing, frequency, modulation, coding rate, etc.). For random access bursts the burst label contents are also required.
- The information comes on one hand directly from the decoded tables in the forward link (DAMA slots) and from the random access management which uses also these tables and other signalling information to produce descriptions for random access bursts.
- **L2-MN:** Interface to monitoring and fault management. On this interface the management module retrieves statistical information and error signals.
- **L2-CF:** Configuration interface. This is used by the terminal infrastructure to configure the RLE module and the scheduler.
- **L2-OB:** Burst output. This interface delivers burst payloads to the physical layer.

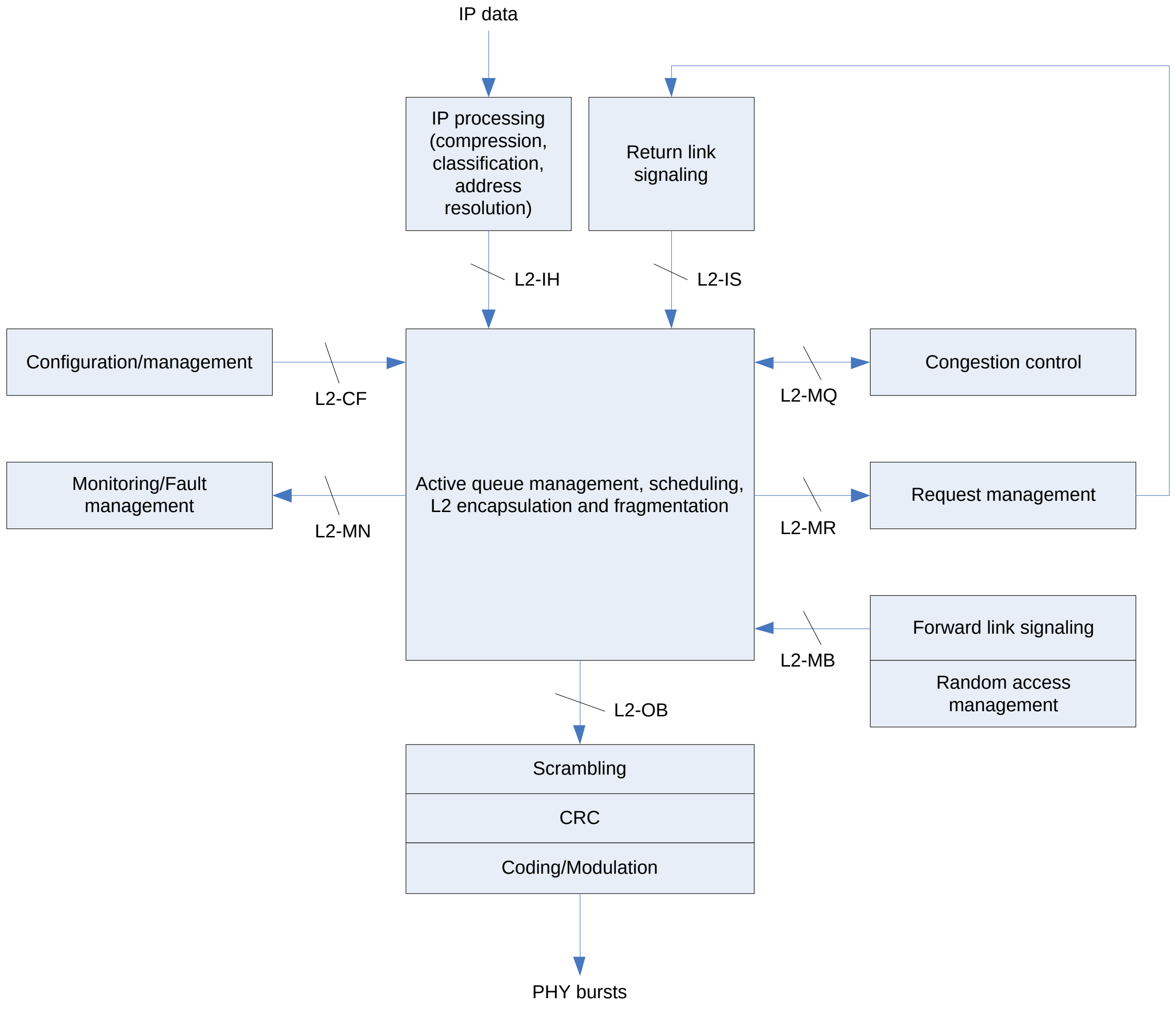


Figure 6.12: RLE transmitter conceptual interfaces

6.4.2.2 RLE transmitter internal interfaces

Figure 6.13 illustrates a conceptual RLE transmitter structure. This structure employs an integrated approach doing joint scheduling; encapsulation and queue management.

The higher layer packets (including signalling packets) are encapsulated and then put into scheduler queues according to their QoS tag. These queues are monitored by the request manager via the L2-MR interfaces and managed by the active queue management which is controlled from congestion control on the L2-MQ interface.

For each possible fragment Id there is one fragmentation context. Each fragmentation context includes at least a fragmentation buffer and the `next_fragment_id` sequence number. The scheduler in each step selects either one of the busy fragmentation contexts or a packet from one of the input queues to produce the next RLE packet. If a packet from a queue is selected and it does not fit completely into an RLE packet, it is put into an idle fragmentation context.

The scheduler is triggered by the burst packing function when it fills a burst. The burst packing function initializes each burst with a signalling byte (if configured to do so) and a burst label (for random access bursts). It then loops around triggering the scheduler until either the scheduler cannot return any data or the burst is full. Then the burst payload is handed over to the L2-OB interface.

The burst packing function is triggered over the L2-MB interface which carries burst descriptions with possibly attached information for the physical layer.

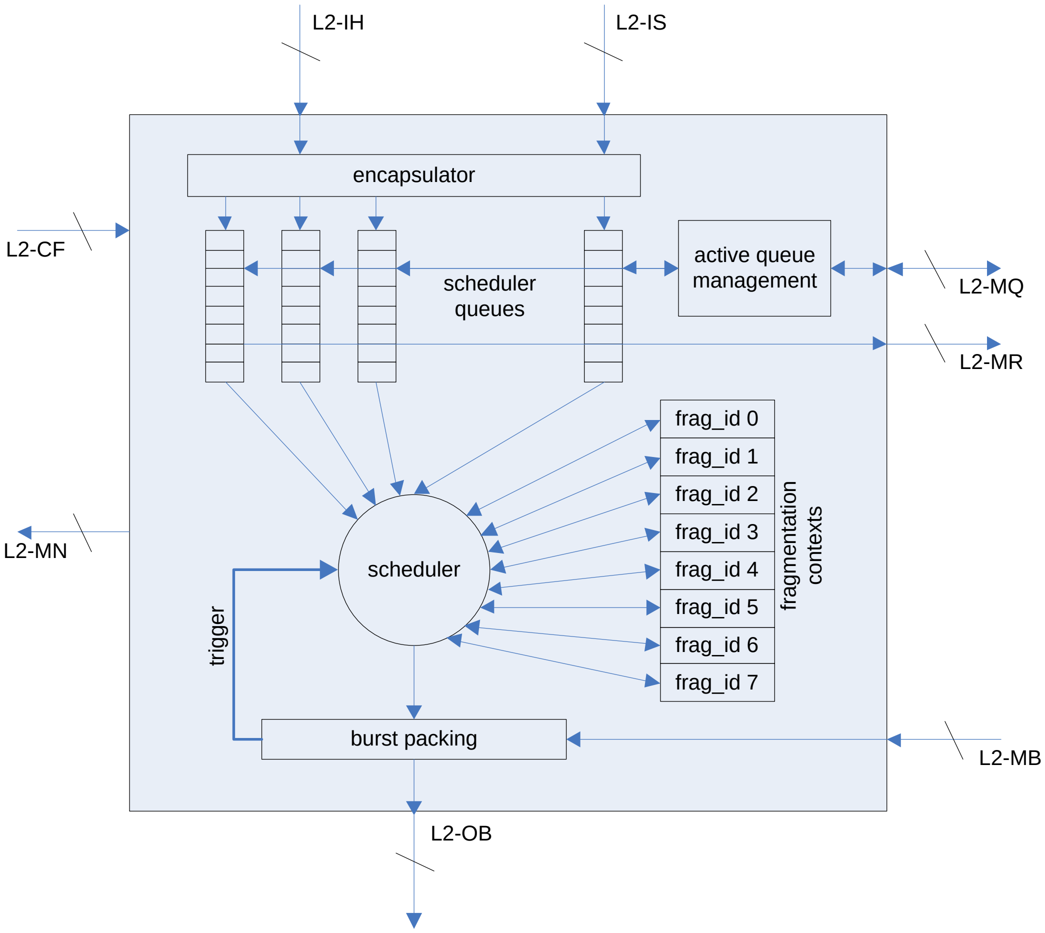


Figure 6.13: Conceptual RLE transmitter structure

6.4.2.3 Internal structure of the receiver

The conceptual structure of the receiver is given in Figure 6.14. The received burst payloads are first split into RLE packets. This can be done by accessing just the Fragment_Length fields of the packets. Unfragmented packets (start and end indicator are both set) are then sent directly to the decapsulation, while fragmented packets are sent to the defragmentation engine. An optional filtering on the fragment label can also be applied here in some meshed scenarios (transparent mesh, fragment switching).

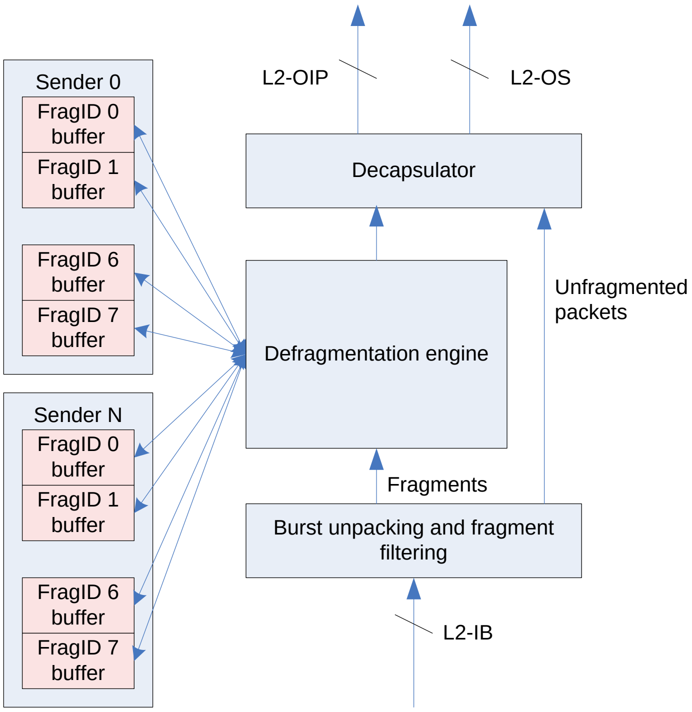


Figure 6.14: Conceptual structure of L2 (receiver)

The defragmentation engine needs to manage fragmentation buffers. Each packet that is currently defragmented requires a buffer which consists of the actual data buffer and some data like the expected sequence number, the CRC flag. The buffers are keyed by the combination of the sender address (which is received together with the burst from the physical layer) and the fragment id - each pair of <sender, receiver> has its own id space. Given that after filtering the receiver should be equal to the terminal; this gives a fragment id space for each sender. For this reason it may be necessary to support dynamic management of these buffers.

Once the packet is defragmented it is handed over to the decapsulator which chops off the protocol type, packet labels and extended headers, possibly inserting default values or decompressing values and dispatches the resulting L3 packet to either the L3 stack or the signalling stack.

The conceptual interfaces of the RLE block are as follows:

- **L2-IB:** On this interface the RLE block receives burst payloads from the physical layer.
- **L2-OIP:** Reassembled and decapsulated L3 packets are sent upstream from the RLE block to the higher layers.
- **L2-OS:** Reassembled and decapsulated signalling packets are sent upstream from the RLE block to the signalling modules.

6.4.3 RLE Implementation Guidelines

The Return Link Encapsulation protocol (RLE) can be sub-divided into three distinct sub-layers as shown in Figure 6.15. The upper layer (encapsulation) takes higher layer packets together with information about extension headers and the associated protocol type (Ethertype) of the packet and produces encapsulated packets. The fragmentation layer takes these and produces RLE packets that contain parts of the encapsulated packets or entire encapsulated packets. The burst packing layer finally produces physical layer burst payloads by taking one or more RLE packets optionally prepending a label and a signalling byte and optionally appending padding and a CRC. These burst payloads are then handed over to the physical layer (PHY) for further processing.

On the reception side the processing is in opposite order: the signalling byte, the burst label, the CRC and padding (if any of them is present) are stripped from the burst payloads by the burst unpacking layer. It then splits off the RLE packets. The fragmentation process reassembles encapsulated packets from the RLE packets and hands them over to the decapsulation where the protocol type, the extension headers and the Layer 3 payload are reconstructed. The result of the decapsulation process is given to the upper layers. The following terminology is slightly different from the terminology in the RCS2 LLS for easier understanding. Table 6.8 provides the mapping between the terms used in the normative document [i.1] and the terms used in the present document.

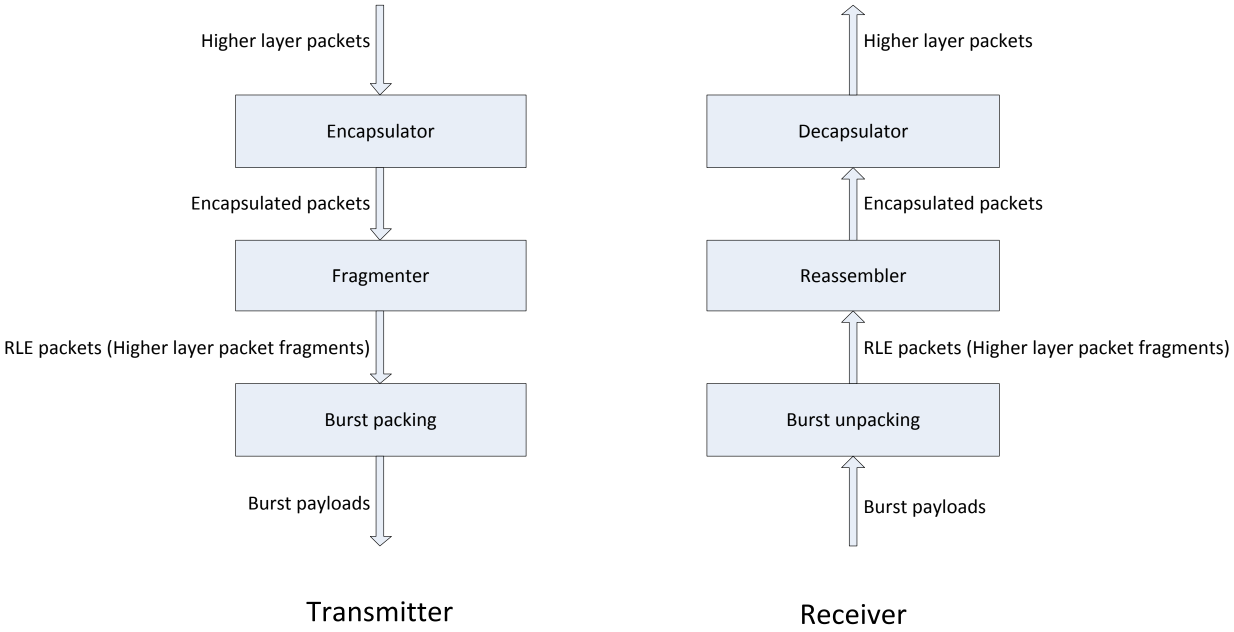


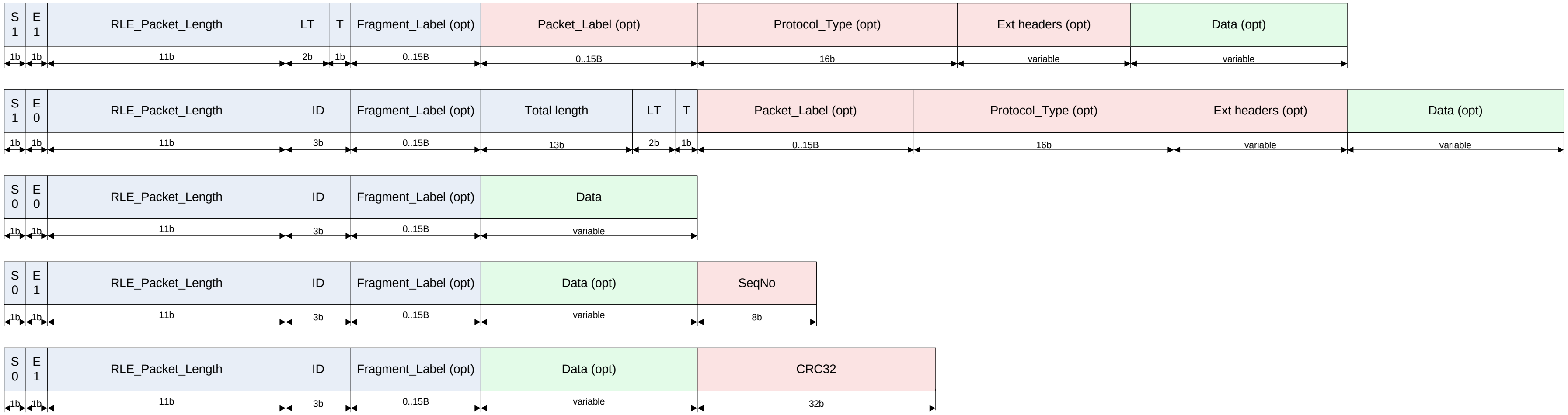
Figure 6.15: RLE sublayers

Table 6.8: Terminology mapping for RLE

LL normative text	LL Section ([i.1])	Guidelines	Comments
SDU	7.1.1	higher layer packet	usually an IP or a signalling packet; the SDU also may include extension headers
ALPDU	7.2.1	encapsulated higher layer packet	higher layer packet with encoded extension headers, packet label and protocol type attached (all of them may be empty)
ALPDU label		packet label	label attached to the higher layer packet (MAC address or SVN number)
PPDU	7.2.2	RLE packet	a packet containing a fragment of an encapsulated higher layer packet or a complete encapsulated packet
PPDU label		fragment label	a label attached to an RLE packet; not used in RCS2 LL; zero length
Frame PDU	7.2.3	burst payload	the payload contents of a physical layer burst
Payload_label		burst label	a label attached to a physical layer burst; can contain addresses or CRDSA information
Payload header map		signalling byte	the optional first byte of the burst payload signalling the length of the burst and fragment labels

Figure 6.16 provides the general format of the RLE packets as specified in the normative document [i.1]. The green fields constitute the original higher layer packet (an IP packet for example). The red fields (except for the sequence number and the CRC) together with the green fields are the encapsulated higher layer packet and the green, the red and the blue fields together constitute RLE packets.

In the RLE burst payload (see Figure 6.17) zero or more RLE packets (green) are concatenated. They optionally may be prefixed with a signalling byte and a burst label. The remaining space in the burst that is not used by RLE packets is filled with padding and the last four bytes may optionally be occupied by a burst CRC (in the normative document [i.1]) this CRC is not used because there is another CRC below the spreading layer). Bit padding at the end occurs if the payload length required by the physical layer is not a multiple of 8 bit. This has been avoided in RCS2 by making all burst definitions have multiple of 8 bits.



S 1	E 1	RLE_Packet_Length	LT	T	Fragment_Label (opt)	Packet_Label (opt)	Protocol_Type (opt)	Ext headers (opt)	Data (opt)
1b	1b	11b	2b	1b	0..15B	0..15B	16b	variable	variable

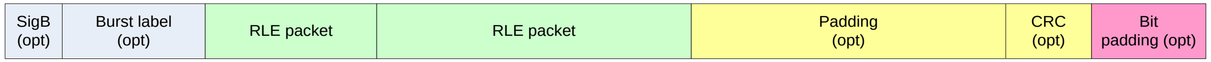
S 1	E 0	RLE_Packet_Length	ID	Fragment_Label (opt)	Total length	LT	T	Packet_Label (opt)	Protocol_Type (opt)	Ext headers (opt)	Data (opt)
1b	1b	11b	3b	0..15B	13b	2b	1b	0..15B	16b	variable	variable

S 0	E 0	RLE_Packet_Length	ID	Fragment_Label (opt)	Data
1b	1b	11b	3b	0..15B	variable

S 0	E 1	RLE_Packet_Length	ID	Fragment_Label (opt)	Data (opt)	SeqNo
1b	1b	11b	3b	0..15B	variable	8b

S 0	E 1	RLE_Packet_Length	ID	Fragment_Label (opt)	Data (opt)	CRC32
1b	1b	11b	3b	0..15B	variable	32b

Figure 6.16: RLE packet formats



SigB (opt)	Burst label (opt)	RLE packet	RLE packet	Padding (opt)	CRC (opt)	Bit padding (opt)

Figure 6.17: RLE burst payload format

6.4.4 RLE Profile Specification

The RLE protocol has a number of features that are configured and used differently for different usage scenarios and different instances in a DVB-RCS2 system. For each of the uses cases the fixed parameters and options as well as the configurable options are collected into an RLE profile. This section provides the RLE profile for a transparent star configuration. It additionally provides some hints and guidelines on designing profiles for transparent mesh and regenerative systems.

6.4.4.1 Transparent star profile

6.4.4.1.1 Higher layer packets

Higher layer packet is completely mapped into the data field of the encapsulated packet. Other means of transporting higher layer packets like the PDU concatenation extension header are not required by the LLS and should be used only when the transmitter knows that the receiver supports this extension header. Examples of higher layer packets are IPv4, IPv6 or signalling packets.

The total length of the higher layer packet, extension headers, the packet label and the (maybe compressed) protocol type cannot exceed 4 095 bytes. The upper layer MTU should be set to a value that takes into account the maximum of 2 byte for the protocol type, the maximum packet label and the maximum length of extension headers that is supported. The maximum packet label length depends on the transmission context (see Table 7-10 in the normative document [i.1]).

NOTE: If during encapsulation of a packet more than 4 095 bytes are produced the packet should be dropped.

6.4.4.1.2 Extension headers

Extension headers are specified in RFC 4326 [i.71], RFC 5163 [i.72] and in EN 301 790 V1.5.1 [i.46]. The list of allocated header fields is defined in the Internet Assigned Numbers Agency (IANA) Next-Header Registry, located at: http://www.iana.org/assignments/ule-next-headers/ule-next-headers.xml. The list of values recommended for DVB-RCS2 is provided in the normative document [i.1].

Extension headers are encoded in the same way as in GSE and are fragmentable.

If the length of the (possibly compressed or defaulted) protocol type, the packet label, the extension headers and the higher layer packet exceed 4 095 bytes, the complete packet should be dropped by the encapsulator.

Care should be taken when using extension headers, especially mandatory headers. The term mandatory in this case does not require that the extension header is supported, but that if the receiver finds such a header and it does not support it, the entire encapsulated packet should be dropped. The LLS does not require support for specific extension headers so other means should be used to determine whether to send extension headers or not.

6.4.4.1.3 Packet label (ALPDU label)

The RLE protocol generically allows configuring the length of the packet label for each of the four label types to have a length from 0 to 15 bytes. For RCS2 three of the label lengths are fixed and the fourth one should be specified by the hub to be 1 byte (Table 6.9).

Table 6.9: RLE label types

Label type	Packet label size	Default protocol type	Comment
0	1	`implicit_protocol_type` of Frame payload format descriptor	type_0_alpdu_label_size of the Frame payload format descriptor
1	3	`implicit_protocol_type` of Frame payload format descriptor	
2	0	`implicit_protocol_type` of Frame payload format descriptor	
3	0	0x0082/0x42 - Internal M&C signalling (L2S)	default protocol type different from label type 2

Label type 3 should be used by the terminal to send signalling information. The default protocol type for label type 3 is the protocol type used for signalling (0x0082 uncompressed/0x42 compressed) and thus can be omitted. The default label type for the other three label types is indicated by the hub in `the implicit_protocol_type` field of the `frame_payload_format_descriptor`.

For normal RCS2 operation label type 0 is used for higher layer traffic on the non-default SVN. In this case the one byte SVN tag needs to be inserted into the packet label. For a higher layer packet on the default SVN the tag may be omitted by using label type 2. Use of label type 1 is for compatibility with on-the-fly translation to GSE and is not mandated by the normative document [i.1].

6.4.4.1.4 Protocol type

The protocol type specifies the type of higher layer packet (for example IPv4) or the type of the first extension header. In order to save bandwidth the protocol type may be defaulted or compressed.

The default protocol types are defined as follows:

- The implicit compressed default protocol type for label type 3 is 0x42 which is the RCS2 internal L2 signalling. This is not changeable.
- The implicit compressed default protocol type for the other label types is indicated by the NCC in the `frame_payload_format_descriptor` field `implicit_protocol_type`.

The protocol type encoding is done based on the values signalled by the NCC as follows (as shown in Figure 6.15):

- The `allow_ptype_omission` flag allows using the default protocol type. If a higher layer packet is to be transmitted and there is a label type which has a default protocol type equal to the protocol type of the packet and a packet label length equal to the packet label of the packet, then that label type should be used for the encapsulation of the packet. In this case the `protocol_type` field of the encapsulated packet (ALPDU) should be omitted and the `protocol_type_suppressed` field in the first RLE packet (PPDU) containing parts or the entire encapsulated packet is set to 1.
- If allow_ptype_omission is not set or there is no suitable label type that allows the omission of the protocol type then, if the `use_compressed_ptype` flag in the `frame_payload_format_descriptor` is cleared the two byte protocol type should be inserted into the encapsulated packet (high byte first) and the `protocol_type_suppressed` field in the first RLE packet (PPDU) containing parts or the entire encapsulated packet is cleared.
- Otherwise if `use_compressed_ptype` is set and the protocol type table contains a compressed value for the protocol type, then the 8-bit compressed protocol type is inserted into the encapsulated packet and the `protocol_type_suppressed` flag in the first RLE packet (PPDU) containing parts or the entire encapsulated packet is cleared.
- Otherwise if the `use_compressed_ptype` is set and the protocol type table does not contain a compressed value of the protocol type the extension mechanism should be used: a value of 0xff is inserted as compressed protocol type and the actual two byte protocol type is inserted (high byte first) directly behind the packet label. The `protocol_type_suppressed` in the first RLE packet (PPDU) containing parts or the entire encapsulated packet is cleared.

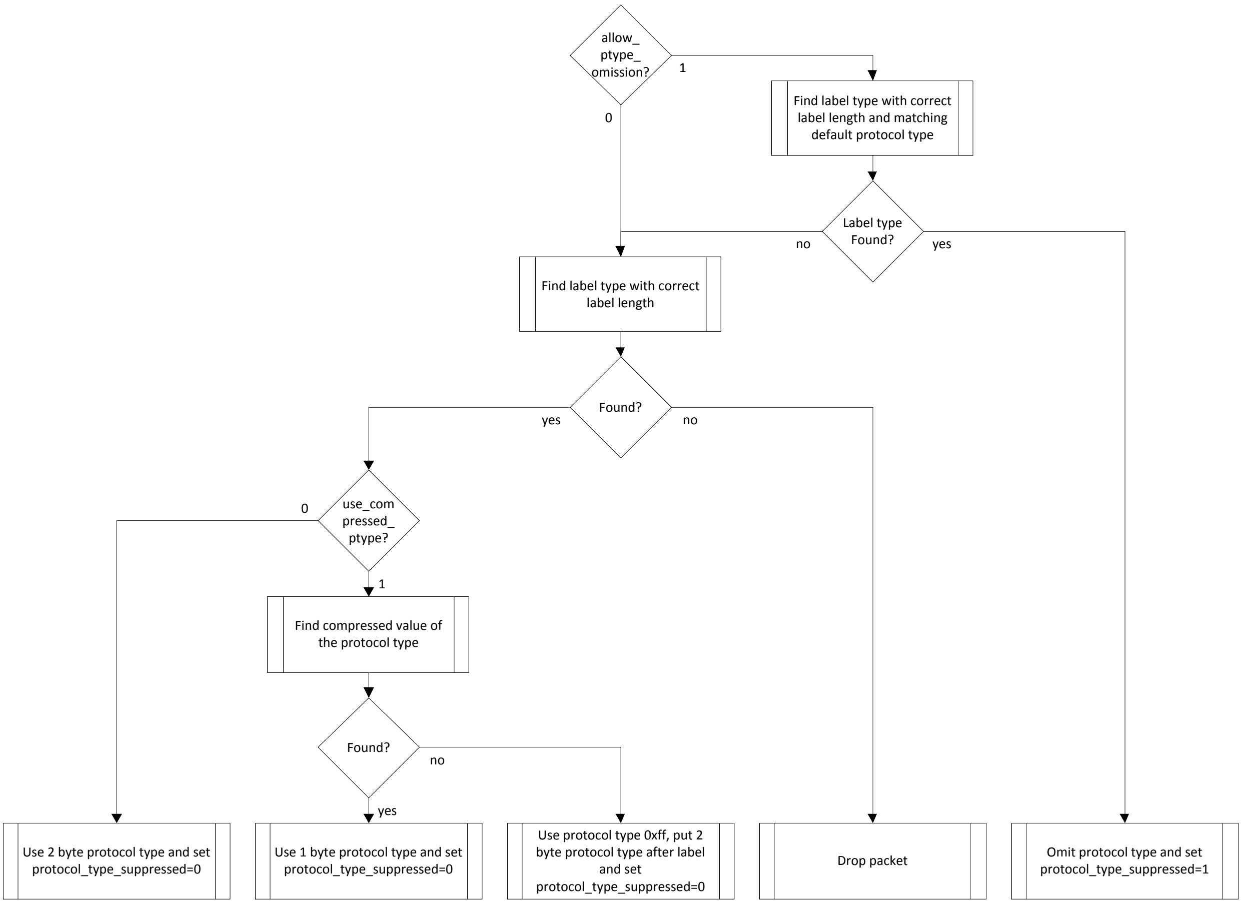

allow_ ptype_ omission?
1
0
Find label type with correct label length and matching default protocol type
Label type Found?
no
yes
Find label type with correct label length
Found?
yes
no
use_com pressed_ ptype?
0
1
Find compressed value of the protocol type
Found?
no
yes
Use 2 byte protocol type and set protocol_type_suppressed=0
Use 1 byte protocol type and set protocol_type_suppressed=0
Use protocol type 0xff, put 2 byte protocol type after label and set protocol_type_suppressed=0
Drop packet
Omit protocol type and set protocol_type_suppressed=1

Figure 6.18: Protocol type encoding

6.4.4.1.5 Total length field

The total length field is the sum of the length of the protocol type (this may be 0 for a suppressed protocol type, 1 for a compressed one, 2 for an uncompressed one or 3 for a compressed using the extension mechanism), the length of the packet label, the length of all extension headers and the length of higher layer packet. The value of the total length may not exceed 4 095. This field is used only if the transmission requires fragmentation of the packet. If the packet is transmitted in a COMPLETE RLE packet, no total length field is used because the packet length can be deduced from the RLE packet length.

6.4.4.1.6 Protocol_type_suppressed field

This bit should be set depending on the protocol type processing (see clause 6.4.4.1.4).

6.4.4.1.7 Label type

This field is set depending on the packet label and protocol type processing (see clauses 6.4.4.1.3 and 6.4.4.1.4).

6.4.4.1.8 Fragment label (PPDU label)

This field is not used and has a zero length for the transparent scenario. This length (0) is signalled via the `implicit_ppdu_label_size` field of the `frame_payload_format_descriptor`.

6.4.4.1.9 Fragment Id

This field is used to mark all RLE packets that contain fragments of the same higher layer packet with the same id so that the packet can be reassembled. The sender should use a round-robin algorithm to use the 8 available fragment ids: when fragmentation is required the search for the next fragmentation id is started from the fragmentation id from the last transmitted START packet plus 1 (in modulo 8 arithmetic). This ensures good spacing of the fragment IDs.

No two currently fragmented higher layer packets should use the same fragment id. This means that a maximum of 8 higher layer packets can be fragmented in parallel at a given transmitter for the same receiver (there is only one return link receiver in a transparent star).

6.4.4.1.10 Sequence number

The use of the sequence number in contrast of the CRC for reassembly error checking is controlled by the NCC in the `frame_payload_format_descriptor`. If the `allow_alpdu_sequence_number` bit is set the terminal should use the sequence number. If the bit is not set, the `allow_alpdu_crc` should be set by the NCC and the terminal should use the CRC field instead. The NCC should set exactly one of these bits.

For each receiver/fragment id pair the transmitter should maintain a variable `next_sequence_number`. For the transparent star case, where there is only one receiver, the transmitter should maintain 8 of these variables (one for each fragment id). These variables are initialized to 0. Whenever the transmitter produces an END RLE packet it inserts the current value of the next_sequence_number for the corresponding fragment id into the sequence number field of the packet and increments the `next_sequence_number` modulo 256. The `use_alpdu_crc` field of the corresponding START RLE packet should be cleared.

6.4.4.1.11 CRC field

If the CRC field is to be used for reassembly error checking (as described in the previous section), the CRC should be computed as follows:

- initialize the 32-bit CRC with the value `0xffffffff`
- feed the following fields into the CRC algorithm (MSByte first LSBit first):
    - the original two-byte protocol type
    - the packet label
    - all extension headers
    - the higher layer packet
- invert the computed value and insert the resulting 4 bytes MSByte first into the CRC field.

6.4.4.1.12 Start_indicator, End_indicator and RLE packet length

The following values are set as required by the fragmentation algorithm in the same way as in GSE:

Signaling byte (payload map)

The use of the signalling byte is controlled by the NCC via `the use_explicit_payload_map` of the `frame_payload_format_descriptor`. The use of this field is not strictly necessary for RCS2, because the fragment label has always the length zero and the burst label has the same length for all burst transmitted from within a given context. Therefore the value of the signalling byte will be the same for all burst in that context.

Burst label

The burst label can have a size ranging from 0 to 15 bytes in RLE. As per the normative document [i.1] however, the sizes are fixed for the DAMA, the slotted aloha and the CRDSA context (see Table 7-10 in [i.1]).

For dedicated access the only method to recover the terminal (source) address of a received burst is implicit recovery by matching the reception time of the burst to the burst-time allocation table. The RCST should produce only the label lengths specified in this table for strict standard conformance.

The `implicit_payload_label_size` allows the specification of the label sizes only for traffic bursts (last row of Table 7-10 in [i.1]); label sizes for the other burst types is fixed and is not signalled.

An alternate method to identify the source terminal which simplifies the hub, but requires corresponding support in the RCST is to explicitly signal the source terminal in DAMA slots. In this case the same label lengths and contents as for the slotted aloha access should be used. Non-standard signalling is required in this case.

Burst CRC

The normative document does not use a burst CRC on the RLE burst packing layer, because a CRC is inserted below the spreading (see Figure 7-8 and clause 7.3.4 of [i.1]).

6.4.4.2 Mesh and regenerative scenarios

This clause provides some general guidelines on how to select profile options for MESH and regenerative systems. The main problem is to select the right format for the burst, the fragment and the packet labels depending on the scenario and the on-board processing technology.

Both meshed and regenerative scenarios require additional labels for (i) addressing and (ii) reassembly.

In a transparent star scenario there is only one receiver on the return link by definition. Therefore, no destination addressing is required at the lower layers. The only kind of destination addressing used is the SVN tag which is (indirectly) transported in the higher layer packet label (ALPDU label). In a transparent star scenario since there are multiple transmitters sending to the single receiver, the receive addressing is required for reassembly. Since all transmitters use the same fragment ID space, the receiver needs to deduce this information, for example, an RLE packet with fragment ID 1 from terminal A should not be reassembled with an RLE packet with fragment ID 1 from terminal B (see Figure 6.19).

For DAMA bursts this is done indirectly by matching the receive time of the burst with the actual transmission plan. This directly leads to the terminal that did send the burst. For random access bursts, the source address is contained in the burst label as specified by Table 7-10 of the normative document [i.1].

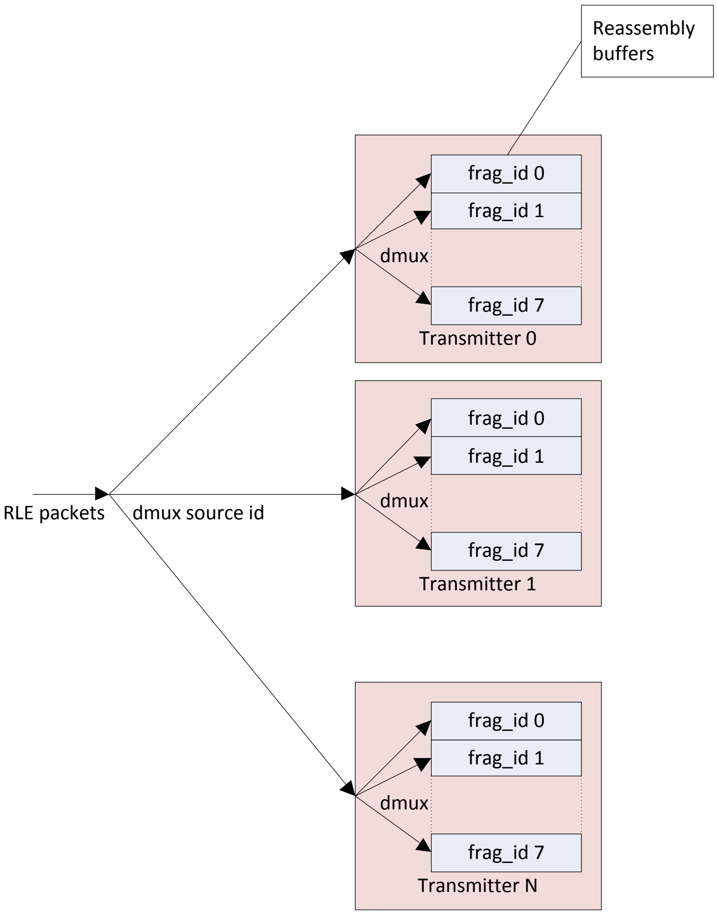


Figure 6.19: Fragment demultiplexing at the RLE receiver

In a mesh system the receiver needs to do the same kind of demultiplexing as the receiver in the transparent star. Source identification can be done in the following ways:

- Implicit by timing. As in the transparent star case the receiver knows the transmission plan and can correlate the receive time of a burst to the plan to find the sender.

- Explicit in the burst label. This may be a MAC address, a logon id or a connection identifier depending on the system.

In a regenerative system the necessity of source identification depends on the switching layer. If the OBP performs the reassembly and fragmentation then there will be only one transmitter received by a given terminal and in this case no source identification is necessary. If the OBP performs RLE to GSE translation or just uses RLE in the downlink source identification is necessary because the downlink will contain RLE packets from more than one sender. The source identification can be done in one of the following ways:

- For GSE in the downlink the only option is to overload the semantics of the fragment id field. The OBP should translate the same RLE fragment id from different terminals to different GSE fragment ids. This maybe done either by using a connection identifier which will limit the number of connections in a downlink to 32 or by dynamically allocating fragment ids and relying on receiver filtering based on the packet label (then a maximum of 256 packets can be switched at the same time into a given down link).

- For RLE in the downlink the fragment label can be used to carry the sender identification in form of a connection identifier or some address.

In a regenerative case a label is also required for switching. This switching can be done on the burst label (in this case all RLE packets in the same burst will go to the same downlink or even to the same destination terminal), on the fragment label or on the packet label. In the latter case the OBP should either do reassembly and fragmentation to switch on encapsulated or IP packets, or it should implement label caching, i.e. read the label from the first RLE packet of a given higher layer packet, enter it together with the fragment id and the source id into a cache and use the cached value to switch all the subsequence RLE packets of this higher layer packet. The different options have different implications in terms of efficiency, overhead, required processing power, required signalling and limits on the number of terminals and connections.

In the case of on-the-fly translation of RLE packets to GSE packets in an OBP the following RLE configuration should be used:

- The label lengths of label type 0 should be set to 6.

- The CRC should be used for reassembly checking.

- The sender should not fragment the protocol type, the packet label (ALPDU label) or the CRC.

6.4.5 RLE Reassembly Error Checking Guidelines

The RLE protocol allows parallel fragmentation and reassembly of several higher layer packets from the same sender. This is achieved by combining the information in the `Fragment_ID` header field with either the `Sequence_Number` field (if the `Use_Packet_CRC` header field is cleared), or with the CRC32 field (if the `Use_Packet_CRC` header field is set to 1). The option to use a 1 byte sequence number instead of a 4 byte CRC32 is provided to reduce the packet overhead from 4 bytes to 1 byte in the cases that the environment allows for that. CRC32 should be used in environments with very large drop-outs (more than 255 bursts in a row, mobile environments, for example) or with OBPs that do conversion to GSE.

It is recommended not to mix both options at a sender. On the one hand, the suitability of one or the other method is a trade-off between propagation impairments, system design (for example in the case of OBPs that do conversion to GSE) and overhead reduction; this allows selecting one method or the other for the sender once. On the other side, letting the sender switch from one to the other option includes higher complexity for scheduling RLE packets in a burst, as the minimum length of the End packet varies for different setting of the `Use_Packet_CRC` header field.

The methods and rules to apply the sequence number and the CRC32 for reassembly error checking are described below.

6.4.5.1 Reassembly error check algorithm with sequence number

If the sequence number is applied for reassembly error check, the following rules should be applied:

- The sender has a variable next_sequence_number for every Fragment_ID from 0 to 7. When emitting an RLE END packet (PPDU) the fragmentation process appends the value of next_sequence_number to the encapsulated higher layer packet and increments the variable. When the value reaches 255, the next increment cycles back to 0.
- The receiver has a variable next_sequence_number for every Fragment_ID. This variable contains the Sequence_Number expected in the next RLE END packet (PPDU). When the receiver receives such a fragment it does the following:
  - It compares the Sequence_Number from the fragment to the next_sequence_number for the given Fragment_ID. If they differ, the reassembled packet is thrown away. If they are equal the packet is passed to the higher layer.
  - It sets the next_sequence_number for the logical link to the value of Sequence_Number + 1 independently of whether the packet was thrown away in the previous step or not. If the new value is 256 it cycles back to 0.

Both the fragmentation process and the defragmentation process initialize their next_sequence_number values for all Fragment_IDs to 0 at initialization time. If the sender sends a packet with a CRC32 the next_sequence_number is not incremented. When the receiver receives a packet with a CRC32, its next_sequence_number is not incremented. This is done so that in the case of the loss of a packet with a CRC32 the next packet with a sequence number is not lost too (a packet with a CRC32 cannot resynchronize the next expected sequence number in the receiver).

Figure 6.20 illustrates how the reassembly error detection mechanism works:

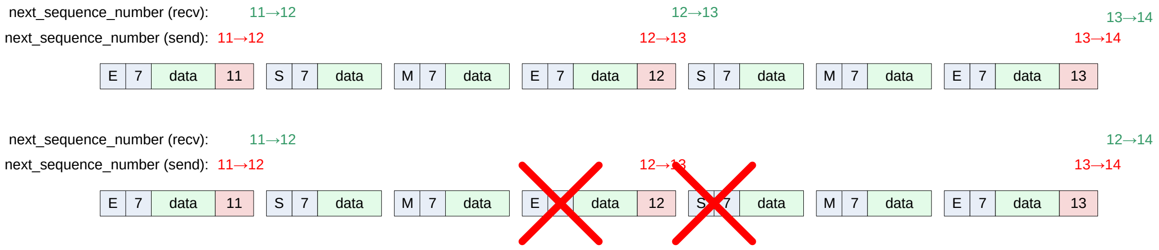


Figure 6.20: Reassembly error detection with sequence number

In this figure the fragmentation and reassembly of several packets on the same Fragment_ID is shown. The upper data flow shows the process without transmission errors and the lower one with two fragments lost (as denoted by the red crosses). The red (lower) numbers show the changes of the next_sequence_number in the fragmentation process, the green (upper) numbers the changes in the defragmentation process.

In the error-less case the first fragment shown is the RLE END packet of a higher layer packet. The current next_sequence_number in the sender is 11 so this value is inserted into the packet trailer and the variable is incremented to 12. The receiver compares the trailer value (11) to its next_sequence_number (11). It finds them to be equal and delivers the packet to the upper layer. Then it sets its next_sequence_number to the trailer value plus 1 (12). The sender then fragments the next packet: an RLE START packet, an RLE intermediate packet and an RLE END packet. It inserts its current next_sequence_number (12) into the trailer and increments the variable to 13. The receiver finds the trailer (12) to equal the next_sequence_number value (12), delivers the packet and sets its next_sequence_number to 13. Then the sender processes the next packet the same way resulting in the next_sequence_numbers on both sides to become 14.

In the error case an RLE START packet and an RLE END packet are lost. It is assumed, that the sum of the sizes of the other fragments happen to sum up to the correct Total_Length value (otherwise the error is detected already by the length check). When the sender sends the second RLE END packet it inserts 12 into the trailer and sets its next_sequence_number to 13. This fragment is lost so the receiver still has value 12. Now the sender sends the fragments for the next higher layer PDU. The trailer now contains 13 and the next_sequence_number is 14. The RLE START packet is also lost (otherwise the error would be detected earlier). When the RLE END packet finally arrives at the receiver it finds a Sequence_Number of 13 which is different from its next_sequence_number (12). Therefore it throws out the reassembled packet and sets the next_sequence_number to the trailer value plus 1 (14).

This mechanism replaces the packet CRC of GSE. The benefit is a reduction in trailer size from 4 bytes to 1 byte. This is only one reassembly error detection mechanism. Other rules for reassembly error detection are:

- If an RLE packet arrives which has the Start_Indicator 0, but no higher layer packet is currently reassembled for the given Fragment_ID, then an RLE START packet has been lost and the fragment is thrown away. If the RLE packet has the End_Indicator equal to 1, the next_sequence_number is updated with the trailer value plus 1.
- If an RLE packet arrives which has the Start_Indicator 1, and a higher layer packet is currently reassembled on the same Fragment_ID, an RLE END packet has been lost. The currently reassembled higher layer packet is thrown away, the next_sequence_number is incremented and the RLE START packet is processed normally.
- If the sum of the length of the received RLE packet and the already reassembled length is larger than the advertised Total_Length then at least an RLE START and an RLE END packet have been lost. Both the currently reassembled higher layer packet and the new RLE packet are thrown away. If the packet has the End_Indicator set to 1 and the Start_Indicator is 0, the next_sequence_number is updated with the trailer value plus 1.
- If an RLE END packet is received and the sum of the Packet_Length and the length of the already buffered packet part does not match the expected Total_Length, the buffered higher layer packet and the new RLE packet are thrown away. The next_sequence_number is set to the trailer value plus one.

6.4.5.2 Reassembly error check algorithm with CRC32

If the CRC32 is applied for reassembly error check, the packet CRC uses the same algorithm as the burst CRC. It is, computed over the 16-bit protocol type (even if suppressed or compressed), the label, the extension headers and the higher layer data of the packet.

The CRC uses the well known CCITT polynomial 0x104c11db7 with an initial value of 0xffffffff and a final negation of the result.

The bit ordering for CRC calculation is as follows: bytes are taken most significant first. Within each byte, bits are taken least significant first. The resulting CRC is appended to the data field of the END packet with the most significant byte first and the highest order bit of the reminder in the most significant bit of the byte.

The following rules should be applied for assembly error detection:

- When emitting an RLE END fragment corresponding to a Fragment_ID, the fragmentation process appends the CRC32 (calculated as stated above) to the higher layer packet data.
- When the receiver receives the RLE END packet of the currently assembled Fragment_ID it does the following:
  - It performs an integrity check by independently calculating the same CRC value over the fields indicated above of RLE packets with the same Fragment_ID as the received RLE END packet.
  - It compares the calculated CRC with the received value in the RLE END packet trailer. If they are not identical (integrity check fails), the reassembled packet with the same Fragment_ID as the RLE END packet is discarded. Otherwise, if the integrity check is successful, the reassembled higher layer packet is handed over to the decapsulator.

Figure 6.21 illustrates how the reassembly error detection mechanism works:

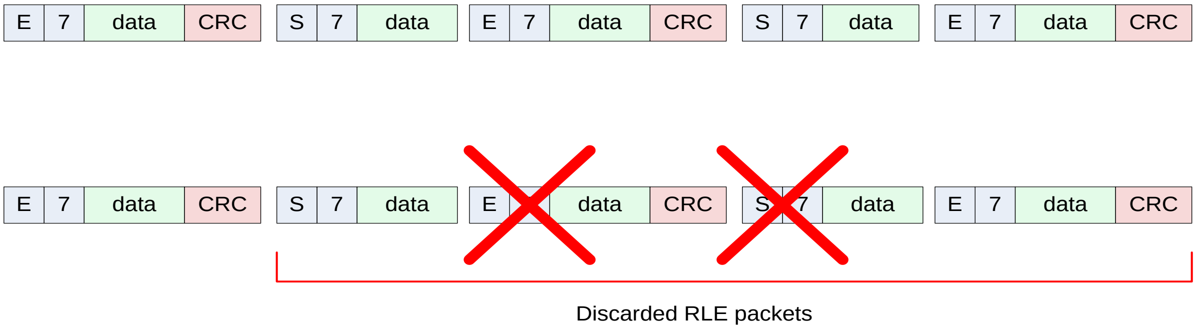


Figure 6.21: Reassembly error detection with CRC32

In this figure the fragmentation and reassembly of several higher layer packets on the same `Fragment_ID` is shown. The upper data flow shows the process without transmission errors and the lower one with two fragments lost (as denoted by the red crosses).

In the error-less case the first fragment shown is the RLE END packet of a higher layer packet (it is assumed that the previous RLE START and RLE intermediate packets are received correctly). The receiver will calculate the CRC and compare it successfully with the received one in the RLE END packet. The same will happen with the two following pairs of RLE START and RLE END packets, so that the reassembled higher layer packets will be handed over to the decapsulator.

In the error case, the RLE END packet of the second higher layer packet and the RLE START packet of the third higher layer packet are lost. This means, that eventually the reassembly function could wrongly interpret that the consecutively received RLE START and RLE END packets belong to the same higher layer packet. If the sizes of the correctly received RLE START and RLE END packets do not match the `Total_Length` field in the RLE START packet, the error would be automatically detected. Otherwise, the CRC mechanism should detect the error by comparing the locally calculated CRC over the reassembled encapsulated packet with the CRC trailer contained in the RLE END packet. If they are not identical, the reassembly function at the receiver will discard both, the RLE START and the RLE END packet.

Other general rules for reassembly error detection (previous to applying the CRC):

- If an RLE packet arrives which has the `Start_Indicator` 0, but no higher layer packet is currently reassembled for the given `Fragment_ID`, then an RLE START packet has been lost and the new RLE packet is discarded.

- If an RLE packet arrives which has the `Start_Indicator` 1, and a higher layer packet is currently reassembled on the same `Fragment_ID`, an RLE END packet has been lost. The currently reassembled higher layer packet is discarded and the new RLE packet is processed normally.

- If the sum of the length of the received RLE packet and the already reassembled length is larger than the advertised `Total_Length` then at least an RLE START and an RLE END packet have been lost. Both the currently reassembled packet and the new RLE packet are discarded.

- If an RLE END packet is received and the sum of the `Packet_Length` and the length of the already buffered higher layer packet part does not match the expected `Total_Length`, the buffered packet higher layer packet and the new RLE packet are discarded.

6.4.5.3 Reassembly process

This clause is an illustration of the reassembly process. It is provided to simplify the understanding of the protocol.

In a transparent system with a star configuration the layer 2 processing in the gateway needs to handle burst streams from all terminals in the system. So there need to be distinct data structures for all the terminals in the system. After demodulation and decoding the resulting burst payload is first process by the upper sub-layer of the physical layer. If this is successful it is handed over to the layer 2. The processes go approximately as follows (implementation details abstracted, star system with no fragment labels assumed):

- (PHY) If the burst is a DAMA burst get the source information from combining the TBTP information with the receive time of the burst and tag the source identifier (`Terminal_Id`) to the burst.
- (L2 burst unpacking/PHY) If there is burst label, separate it from the data and process it (for example for random access). If the burst is a random access burst gets the source `Terminal_Id` from the label and tag it to the burst.
- (L2 burst unpacking) Set a pointer to the start of the packet (after the label) and loop until the pointer points beyond the burst or at the last byte: If only one byte is left this byte is a padding, therefore the burst is already processed:
    - (L2 burst unpacking) From the next two bytes decode the `Fragment_Length` field. If this is zero, this is either the start of padding (with `Start_Indicator` and `End_Indicator` both zero) or an error. In any case this burst is done.
    - (L2 burst unpacking) Add 2 and the `Fragment_Length` to the current pointer. This new pointer will be the start of the next RLE packet used for the previous step in the next loop.
    - (L2 burst unpacking) Decode the `Start_Indicator` and the `End_Indicator`:
        - (L2 burst unpacking) If both indicators are 1 this is a full RLE packet so hand over the entire payload of the RLE packet, the last four bits of the header (the `Label_Type` and the `PType_Suppressed` flag) and the `Terminal_Id` to the decapsulator.
        - (L2 burst unpacking) Hand over the packet payload, the last four bits of the header (the `Fragment_ID`), the `Start_Indicator`, the `End_Indicator` and the source information (`Terminal_Id`) to the reassembler for the given `Fragment_Id` and terminal id. Each combination of `Fragment_Id` and terminal id defines one reassembly context storing all the data for a single packet reassembly (`next_sequence_number` variable, reassembly buffer and others).
    - (L2 burst unpacking) Set the current burst pointer to the pointer computed above and repeat the steps for the next RLE packet.

The reassembler is basically a switch statement over the Start_Indicator and End_Indicator values:

- Locate the reassembly context for the given combination of Terminal_ID (source information) and Fragment_ID. This context contains among other fields the reassembly buffer and the next_sequence_number field and is responsible to handle packets for one Fragmentation_ID of a single sender terminal.
- If `Start_Indicator`=0 and `End_Indicator`=0 then (intermediate packet):
    - If `Use_Packet_CRC` is set to 0:
        - Do the checks described in clause 4.1 and if they are ok append the fragment payload to the reassembly buffer.
    - If Use_Packet_CRC is set to 1:
        - Do the checks described in clause 4.2 and if they are ok append the fragment payload to the reassembly buffer.

- If Start_Indicator=1 and End_Indicator=0 then (START packet):
    - Ensure that the RLE packet length is at least 2 and decode the Total_Length, the Label_Type, the Type_Suppressed flag and the Use_Packet_CRC flag.
    - If Use_Packet_CRC is set to 0:
        - Do the checks from clause 4.1 and if they are ok allocate the reassembly buffer (if the implementation strategy requires this). Put the fragment payload into the reassembly buffer and remember the Label_Type, Type_Suppressed, Use_Packet_CRC flags and the Total_Length.
    - If Use_Packet_CRC is set to 1:
        - Do the checks from clause 4.2 and if they are ok allocate the reassembly buffer (if the implementation strategy requires this). Put the fragment payload into the reassembly buffer and remember the Label_Type, Type_Suppressed, Use_Packet_CRC flags and the Total_Length.
- If Start_Indicator=0 and End_Indicator=1 then (END packet):
    - If Use_Packet_CRC is set to 0:
        - Do the checks from clause 4.1 and if they are ok append the fragment data to the reassembly buffer. Chop of the Sequence_Number from the buffer end and compare it with the next_sequence_number. If they are equal hand over the payload, the Label_Type field, the PType_Suppressed flag and the Terminal_Id to the decapsulator. Set the next_sequence_number to the value from the packet plus 1.
    - If Use_Packet_CRC is set to 1:
        - Do the checks from clause 4.2 and if they are ok append the fragment data to the reassembly buffer. Calculate locally the CRC over the fields indicated in clause 4.2 of the RLE packets in the reassembly buffer. Compare the result with the CRC trailer in the RLE End packet. If they are equal hand over the payload, the Label_Type field, the PType_Suppressed flag and the Terminal_Id to the decapsulator.
- If Start_Indicator=1 and End_Indicator=1 then (full packet):
    - This should not happen because this was processed at the burst unpacking layer

The decapsulator receives the payload, the Label_Type field, the PType_Suppressed field and the Terminal_Id. Depending on the two flag values it parses the payload into the label, the protocol type (if necessary taking the default protocol type) and the extension headers and hands over everything together with the Terminal_Id to the higher layer functions.

Conceptually there are three sublayers in this process: the lowest layer handles the packing and unpacking of RLE packet into and from bursts (including burst label and signalling byte if necessary), the middle layer handles just reassembly and needs to look only at the Terminal_ID, the two bytes from the fixed header, the first two bytes of the fragment payload in RLE START packets (the total packet length) and at the trailer in the RLE END packets. All other data is just appended to the buffer transparently. Once the packet is reassembled it is passed to the upper sublayer which does higher layer packet processing: getting the protocol type field, handling extended headers and the optional label. This mechanism also allows the header fragmentation, because the only variable header field actually needed during the processing of the reassembly sublayer is the total packet length.

6.5 Demand Assignment in DVB-RCS2

An RCST using dedicated access should send capacity requests to the NCC for the traffic to be sent on the DA-AC. These requests are made by the RCST using a Capacity Request (CR) message sent to the NC. The CR messages are generated by a Request Class (RC).

A RC describes a method to provide allocations for a DA-AC. An RCST may use more than one RC, each identified by an RC identifier (RC_INDEX). This RC_INDEX is included in the CR message sent to the NCC.

On reception of a CR, the NCC makes the corresponding allocations, taking into consideration any information associated with the RC (such as relative priority of the CR or limits on resource usage). A set of corresponding allocations is made by the NCC in the TBTP2. These allocations are associated with a specific DA-AC, identified by mapping the `assignment_id`.

The RC may be accessed from the HLS Service through an LL Service. One way to realize the required interaction is to feed information about the HLS PDU queue (BA) from the HL Service to the LL Service. This information could include the queue size and average BA arrival rate. The LL Service could then use this information to generate a CR for the BA, or a set of BAs that correspond to the same RC. An LL Service is expected to use a RC that is compatible with the QoS expectations of the HL Services that they support. This requires configuration of a set of RCs and a mapping between the BA (or its PHB) and the set of available RCs.

The behaviour of an RC is defined by its usage of the set of capacity categories: CRA, RBDC, A/VBDC, FCA (defined below). Each RC can support any mix of capacity categories. One of three different sets of request options can be authorized for each RC. The mapping of the capacity category to a RC is configured for the RCST.

The allocation process can support the following six allocation methods:

- Constant Rate Assignment (CRA): Rate capacity, which is provided in full for each allocation while required. This capacity is not requested using a CR but may be associated with an RC.
- Rate Based Dynamic Capacity (RBDC): Rate capacity, requested dynamically by the RCST using an RBDC CR under a specified RC.
- Volume Based Dynamic Capacity (VBDC): Volume capacity, requested dynamically by the RCST using an VBDC CR under a specified RC. These requests are cumulative (i.e. each request adds to all previous requests from an RCST).
- Absolute Volume Based Dynamic Capacity (AVBDC): Volume capacity requested dynamically by the RCST. The VBDC capacity is provided by the NCC in response to explicit CRs from an RCST. These requests are absolute (i.e. replace any previous VBDC CR).
- Random Access Capacity (RA). Capacity assigned to a set of RCSTs for shared access. This capacity is not requested using a CR.
- Free Capacity Assignment (FCA): Volume capacity assigned to an RCST from the free capacity that would be otherwise unused. This capacity is not requested using a CR.

The behaviour of each RC in terms of the type and amount of capacity requested is configurable and may use for any desired combination of (A)VBDC, RBDC and CRA. The default behaviour of the default RC class is to use VBDC for any amount that exceeds the CRA allocation. If the NCC authorizes use of the volume-based capacity category, it will offer AVBDC and VBDC in combination, to an RCST or an RC. The combination of AVBDC and VBDC is seen as a single Capacity Category, denoted A/VBDC. CRA and FCA can be combined with any of the above.

6.6 Examples of QoS configuration

The RCST builds a set of CRs for each RC using DA. This is achieved through the coordination of HL and LL Services. LL Services provide mechanisms to access satellite resources, such as RA or DAMA. HL Services may be used to implement IP services, such as traffic queuing and support for PHBs and are mapped to LL Services.

The characteristics of the available RCs should be considered when selecting a mapping for a HL Service to realize a DiffServ PHB. An HL Service implementing the EF PHB requires a bandwidth guarantee and is typically used to provide low loss, low latency and low jitter service. EF traffic tends to vary slowly. CRA is a suitable option to meet these requirements. RBDC may be used to supplement the CRA, but it may not be desirable to use this alone because the delay to obtain an RBDC assignment may not meet the low latency requirement. The rationale behind of using RBDC to supplement CRA is its offer a higher efficiency: users may tolerate the increase of delay and delay variation due to RBDC in exchange for a more efficient allocation. Since EF traffic requires low jitter, the HL Service may assign this traffic to a higher scheduling priority and use a SA that allows pre-emption of jitter-tolerant traffic.

6.6.1 Example of support for basic DiffServ QoS

An HL Service that implements the AF PHB demands a delivery guarantee but does not mandate constraints on delay or jitter. Since AF traffic may be highly variable, dynamic bandwidth allocation methods are recommended. An RC that supports RBDC may be suitable to meet the AF requirements, since it offers a bandwidth guarantee (up to a certain level) and can efficiently handle bursty traffic. VBDC may also be used to supplement the capacity requested via RBDC. Although VBDC may be assigned with lower priority in some systems, it can respond to a request with high efficiency. Use of VBDC may allow a trade-off between offered QoS and the cost of capacity. Since AF traffic does not require low jitter, the HL Service may assign this traffic to a scheduling priority that is lower than EF and use a SA that allows pre-emption of this traffic by jitter-sensitive traffic.

An RCST may implement several HL Services offering AF support, or variants of the HL Service adapted to the needs of specific classes of traffic. The variants may be mapped to the same SA.

An HL Service that implements the Best Effort PHB has no specified requirements. Consequently, BE traffic may be mapped to an RC that uses VBDC only. Scheduling should ensure that this class does not take resources away from EF or AF services, and traffic should not share the same SA as used for EF.

An RC may be mapped to multiple Capacity Categories, in which case a weight may be applied to the different categories. Different studies have been performed to reach the best way to map capacity categories into PHBs. Some example request strategies are proposed below, based on periodic assignments:

EF = 80 % CRA + 20 % RBDC

AF = 70 % RBDC + 30 % VBDC

BE = 100 % VBDC

Using the above mapping, an RBDC rate may be requested that is equal to 20 % of the ingress rate in the EF queue + 70 % of the new ingress rate in the AF queue.

Possible mappings between allocation channels and SAs are:

- A mapping that assigns all allocated timeslots to a single SA.
- A mapping that provides strict separation between a set of SAs.

Assuming the first type of mapping, the RCST will use the HL service information to identify the relative priority between each BA (i.e. BA queue priorities) that is mapped to the same SA.

As an example, the scheduler could assign the highest priority to PDUs queued in a BA with an EF PHB (to satisfy low latency requirement), followed by the BAs with an AF PHB, and finally the BE traffic, which may receive the lowest priority. Any portion of the total assigned capacity that is not used by a higher-priority queue would be available to a lower-priority BA. This optimizes the use of the according LL service.

In this example, a higher-priority queue can hence use capacity allocated in response to a previous CR for a lower-priority queue, allowing it to serve the new incoming higher priority traffic with less delay. However, this creates a temporary condition of under-allocation for the lower-priority queues (observable as increased delay for the BAs assigned to these queues). The queued lower priority traffic will be sent when capacity is allocated in response to the CR for the high priority CR. Traffic conditioning may be used to control the amount of traffic admitted to a BA with the EF PHB to control this "capacity borrowing".

6.6.2 Example 1 - RCST configuration

The diagram in Figure 6.22 shows an instantiation of the general QoS model. It illustrates the relationships between modules and identifies the HL QoS functions and the LL QoS functions. Solid lines represent the flow of PDUs and other data through the system, whereas dashed lines are used to denote control relationships. Hexagons represent functions and rectangles represent QoS objects.

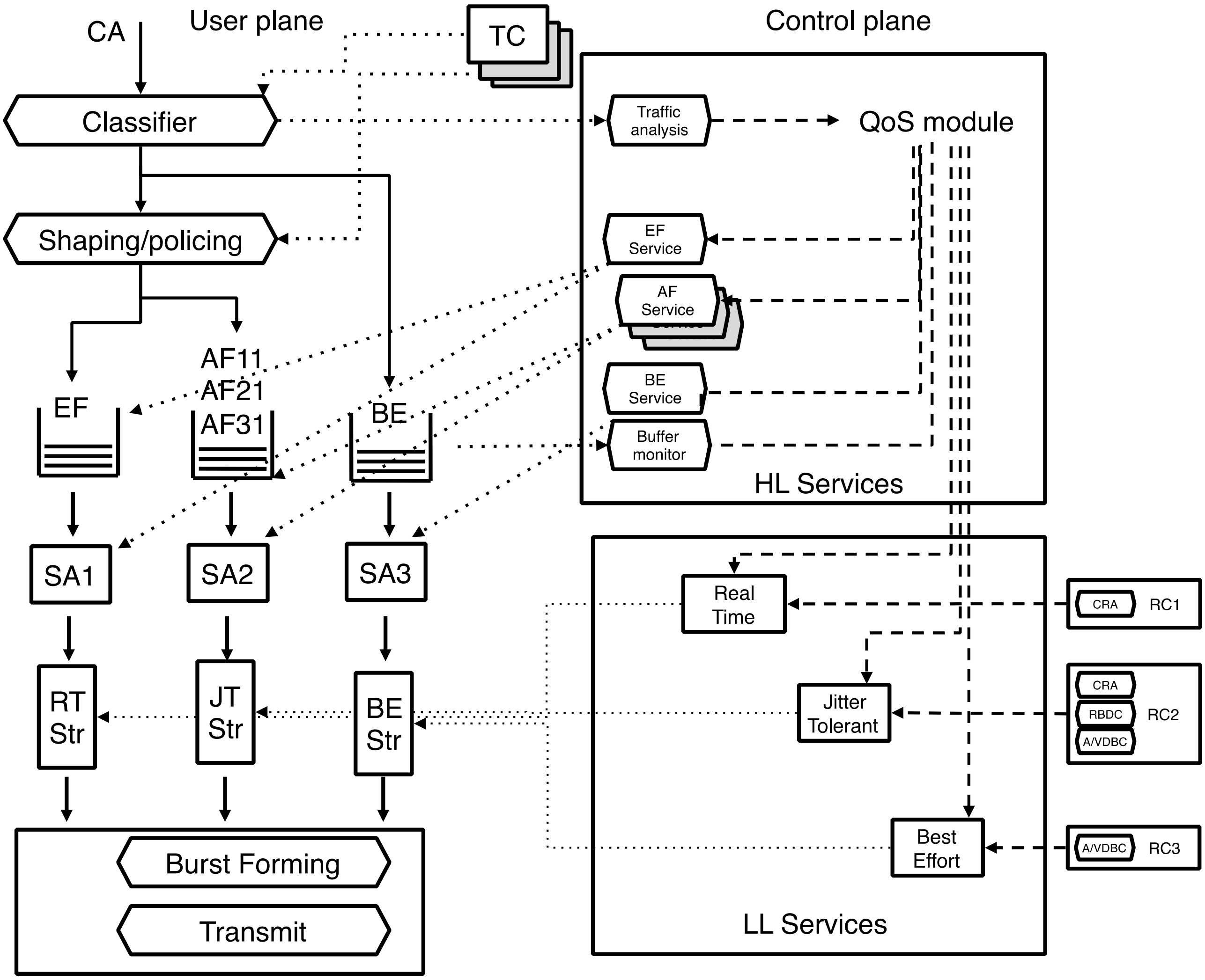


Figure 6.22: Example of QoS model instantiation

In this example, IP traffic arriving at the LAN interface of an RCST is routed to a CA. The CA is classified into four BAs according to their DSCP code-point:

- EF traffic is mapped to a BA HLS PDU queue, managed according to the EF PHB.
- AF traffic is mapped to one of two BA HLS PDU queues, managed according to the AF PHB.
- BE traffic is mapped to one BA HLS queue, managed according to the BE PHB.

Three HL Services in the control plane are instantiated to handle the set of BAs: EF, AF and BE. The HL Services are mapped to three distinct LL Services: the EF to the Real Time (RT) service, the AF to Jitter Tolerant (JT), and DF traffic to the Best Effort (BE) LL Service.

Each LL Service is characterized by its respective RC. Table 6.10 lists recommendations for mapping rate parameters for PHBs to DVB-RCS2 capacity limit values to derive the configuration parameters for the RCs.

Each BA (HLS PDU queue) is mapped to a SA. A scheduler operates on the traffic forming a SA. The scheduler may be triggered each transmission opportunity (notified by the TBTP2) to select the PDUs to be segmented/encapsulated into the LL stream. In this case, a set of SAs are considered, the selection of PDU is based on HL Service parameters and LL Service information. The mapping to a LL Service ensures that PDUs or segmented PDUs are sent using the corresponding AC. When required, PDUs pass through a segmentation function, so that any unsent data is postponed to a later scheduling opportunity. Each segment from an SA is then encapsulated the corresponding configured stream (Str in the diagram) and is then placed in the burst for transmission. This scheduler could, for example, use a strict priority scheduler or a weighted priority algorithm. In this example, three LSs are configured: the RT stream can pre-empt the nRT or BE streams.

The example configuration parameters for the HL Services are summarized in Table 6.10. A set of TCs defines the mapping of traffic to the corresponding HL Service. The TC may optionally define conditioning parameters (e.g. in terms of bandwidth or rate). The assumption is that only two rate parameters are used, as a part of an IP flow profile - the Assured Rate and the Peak Rate, the meaning of these parameters is dependent on the HL Service.

Table 6.10: Example of QoS bandwidth parameters

DVB-RCS2 PHB	DVB-RCS2 RC for HL Service	RC Capacity Category parameters	Comments

EF PHB	Real Time	CRA = ΣAssured rate
		RBDCMax = 0
		VBDCMax = 0
			CRA is the guaranteed IP flow rate

AFn PHB	Critical Data	CRA = 0
		RBDCMax = γΣGuaranteed rate
		VBDCMax = γΣ(Peak - Guarantee)rate
			If there is assured rate, it can be mapped to CRA.
			RBDCMax is a fraction (γ) of the sum of the guaranteed rate
			VBDCMax is a fraction (γ) of the sum of difference between the guaranteed and peak rates.

BE PHB	Best Effort	CRA = 0
		RBDCMax = 0
		VBDCMax = δΣSDR'
			VBDCMax is a fraction (δ) of the total peak rate.

NOTE 1: All parameters apply to the uplink return path, therefore they are transmit parameters. NOTE 2: Assured and Peak Rate are per aggregate rates requested by an RCST. NOTE 3: With the exception of AFn services, all other services map to a single RC. In the case of an AF services, several HL Service instances (n=1-4) are typically mapped to one RC. Therefore, the RC capacity limit values should reflect the aggregation (sum) of individual rates, possibly with an allowance for statistical multiplexing. NOTE 4: The sum of all capacity limit values for all request classes should not exceed the RCST transmission rate.			

6.7 Recommendations for the use of Request Classes

Each RCST is configured to support at least one RC, with some combination of capacity categories. The set of methods is authorized by NCC configuration and NMC management via a MIB. An RCST may be allowed to use several RCs. This requires the RC to also be configured with a set of TCs to classify the PDUs that arrive on the LAN Interface, associating each with one RC. The associated RC can use a NCC-assigned combination of capacity categories. To formulate a CR for the RC, the RCST measures a set of parameters to assess the capacity required. Examples include: the volume of queued traffic associated with the RC, the arrival rate at the RCST, and the transmission rate. This allows the RRM to provide a capacity requirement estimate in terms of additional/absolute volume and rate.

An RC that supports several capacity strategies allows the RCST to make policy decisions about the ratio of capacity that it requests using each capacity category. For instance, both RBDC and VBDC methods may be used to request an allocation that corresponds to a certain number of timeslots per second, but may result in different allocation patterns, different costs, and different reactions to congestion.

The NCC will usually associate resource limits with each RC (this may be configured via a MIB). The DVB-RCS2 Lower Layer Specification places limits on the requested level of resources.

A simple instantiation may statically map one type of traffic to a specific RC and also associate this with a particular HLS Service. However, in general, there is not a need for a 1-to-1 mapping between use of RCs (for resource management) and an HL Service (for QoS management).

Each CR message is identified by the RC (via the rc_index) when it is sent to the NCC. The NCC may in addition associate additional parameters with a specific RC (e.g. priority for allocation relative to other RCs, or timing requirements for allocated slots). In this way, time slot allocation process at the NCC may be tuned for a specific RC.

There are several ways that RCs may be used in the RCST. Examples of use include:

- In RCSTs that perform traffic classification (e.g. QoS based on DSCPs), an RC may realize a specific RRM scheme tailored to a traffic class. For instance, web interactive traffic may be requested using rate-based method (RBDC) while bulk and low priority traffic could use volume-based methods (VBDC). The allocation methods at the NCC may be designed to maximize utilization when several traffic classes are simultaneously used, and a combination of capacity categories may be more optimal than using one alone.
- RCs can also be used to identify a dedicated resource for control signalling and could for example be used to protect assign a higher resource priority for signalling exchanges or to accelerate the control signalling exchange (e.g. decreasing time for session setup when an RCST is idle).

- RCs can be used to virtualize the underlying service. They can provide hard segregation/isolation of flows assigned to different SLAs operating over the same RCST equipment. This is useful when a satellite interface is shared among several users or ISPs. Each ISP/user makes CRs using their own RC for the capacity they need, subject to their own SLA. Allocations by the NCC are identified so they may be associated with the corresponding RC. A refinement of this scheme allows unused capacity at an RCST to be made available in an allocation pool that allows other RCs to use this to maximize their performance.

An NCC may associate a correspondence between an RC and allocation parameters set at the NCC. This allows an RCST to influence the allocation for specific flows. Examples of use include:

- An RC may allow the allocations to be tuned to realize a maximum allocation delay.

- An RC may indicate that predictive-allocation (beyond the rate requested in a CR) is useful and hence influence FCA allocation, allowing the NCC to preferentially allocate unassigned capacity to the RCST that use these RCs.

- An RC may be used to specify a pattern of allocation within the frame (e.g. to spread allocations over time). For example, an NCC may support "feathering" of voice timeslots across a frame, instead of allocating all capacity in one large timeslot burst. This minimizes jitter by delivering voice packets smoothly and evenly in systems using a large Frame period (e.g. hundredths of ms), reducing the time traffic needs to be held in jitter buffers by VoIP gateways.

- RCs may be used to implement a resource priority scheme. This protects capacity should the satellite resource become congested, ensuring that premium services receive allocations ahead of other services. This may also determine which services may be delayed or not allocated in the case of RRM congestion. Prioritizing flows only has benefit when the load is high, under normal conditions all RCs will receive allocation.

Table 6.11 summarizes some examples of RC use, the way in which the RC is applied, and the benefits that may result.

Table 6.11: Example uses of Request Classes

RC usage type		Applications	Benefits
DiffServ resource management	Dynamic allocation	IP QoS. Implicit classification of traffic based on PHB.	Capacity allocated based on CRs that use capacity category tuned to traffic specification.
	Beyond dynamic allocation	Associates extra params at NCC for the RC, e.g. to enable FCA (or set a CRA rate) for a minimum rate guarantee.	Capacity allocated based on CRs, but NCC may provide more (e.g. a minimum guarantee for an RCST during low traffic.)
Integrated Services resource management		Dynamic RC configured to match IS request.	NCC tunes RC properties based on signalling rather than CRs.
Protection from congestion of satellite resource		Protection for Premium Services. Protection for user categories.	Premium services or groups of users are not degraded during resource congestion.
Service Virtualization	Hard resource partitioning	SLA separation for different services at one RCST. Requires allocated timeslots to reference an RC.	RCs permit creation of multiple SLAs on same physical link. Separation between capacity for each SLA.
	Reusing allocated capacity	Link sharing among SLAs. Requires allocated timeslots to reference an RC.	May share unused capacity, to increase user performance.
Satellite Link management	Signalling	Capacity reservations for control flows. Separate management, Accounting.	Management can be classified in a specific RC to reserve bandwidth for out-of-band signalling.
Feathering (spreading burst allocation across the superframe)		Associates extra params at NCC for the RC.	Provides a smooth allocation (rather than bursty) for selected flows.

6.8 Joint use of RA and DAMA access mechanisms

This clause provides implementation guidelines material for the integration of Random Access Allocation Channels (RA-AC) and Dedicated Access Allocation Channels (DA-AC).

The section first presents a number of used for user traffic transmission on the return link via random access.

Then, the rest of the section presents two different means of integration:

1) Integration of DA-AC and RA-AC for user data transmission.

2) Integration of DA-AC and RA-AC for user data transmission and signalling transmission.

6.8.1 RA use cases

6.8.1.1 RA cold start

RA Cold Start is the simplest scenario, when an RCST is logged onto a network, but is initially idle with no current (or infrequent) capacity allocations. This includes infrequent CRA control slots (e.g. following a period with no traffic from an RCST).

If new traffic arrives while the RCST is in this state, the DA request-allocation cycle can introduce a significant delay before transmission. In addition, when traffic is still growing (e.g. Following a UDP DNS packet or TCP SYN), it is difficult to accurately predict the size of CR needed for the initial demand in terms of volume, rate and required time of allocation. This can result in DAMA slots introducing delay (when optimizing for allocation efficiency) or being allocated over-allocated (when optimizing for RCST performance).

In Cold Start, the RA channel could be used to start transmission, until capacity becomes available using the DA channel. Traffic could then be switched from the RA to DA.

6.8.1.2 RA-DAMA top-up

The Top-Up use-case arises when there is a sudden (unpredicted) increase in traffic. This could occur due to a scene-change in video, or the start of additional traffic flows. This use-case would use the RA channel to temporarily provide extra capacity (top-up) until additional requested DA capacity is received or the traffic burst passes. This use-case may mitigate the impact of jitter and/or sudden changes in demand.

6.8.1.3 RA-DAMA back-up

A variant of Top-Up arises when allocation varies (e.g. due to rain-fade, system load, or mobility movement, or loss of a CR). In this case an RCST could use a combination of RA and DA capacity. The usefulness of this case will depend on many factors including system dimensioning.

6.8.1.4 RA IP queue

The impact of DAMA access delay depends on the type of applications using the service. The response time for short interactive applications is a key performance index with a recommended value of 0,5 seconds per page for interactive gaming and 2 seconds for web browsing [i.68] and [i.69].

Table 6.12: End user performance expectations

Applications	Target Response Time
E-commerce, ATM	4 seconds
Web Browsing	2 - 4 seconds
Telemetry, Interactive Games	0,5 seconds

QoS techniques allow a separate IP queue behavior to be assigned for these applications or traffic types. Typical Internet QoS methods (e.g. the Differentiated & Integrated Services model) are designed to avoid significant re-ordering of packets within a flow. That is, they attempt to preserve per-flow order. These methods do not produce unusual pathologies when used with TCP.

6.8.1.5 RA capacity requests

The RA channel could be used to send CRs more rapidly than relying on CRA or periodic control slots. This technique can be used singly or in conjunction with another method (This is not examined here).

6.8.1.6 RA for SCADA

There are at least two distinct types of SCADA that may be classified as Managed SCADA (scheduled polling for data) and Random SCADA (e.g. alarms and events triggered by data). One of the advantages of RA in this domain is its ability to accommodate the large number of terminals required in many SCADA systems. Unlike DA, the complexity of using RA does not increase rapidly with increasing number of terminals. Hence the network size may be only limited by the total offered traffic. RA is also suited for applications that require transmission of short packets.

6.8.2 Integration of DA-AC and RA-AC for user data transmission

The diagram reported in Figure 6.23 provides a conceptual summary of the key steps for transmitting data over a return channel. Traffic arrives in the form of IP datagrams to a Traffic Management Policy module, which applies classification based on some criteria, such as QoS, packet size, resource availability, or the inferred congestion level over the RA-AC. Classified data are then forwarded to a scheduling block and enqueued in L3 buffers for transmission. The scheduler monitors the state of the queues, generating reports (periodically or event-driven) that can be used for requesting capacity from the NCC, as in the case of traffic to be sent over a DAMA channel. Independently, an encapsulator triggers the scheduler at intervals, e.g. once per each superframe, asking for data. Scheduling requests are generated based on the amount of resources available for transmission at the terminal, and result in the scheduler forwarding a proper number of SDUs to the Encapsulator, which in turn fragments, encapsulates and passes them to the modem.

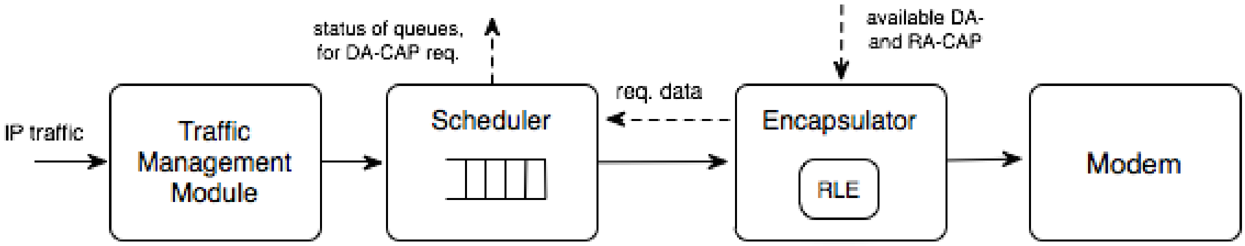


Figure 6.23: Generic traffic transmission scheme for the return link at a terminal

If data packets can only be sent over the DA-AC, as in 1st generation DVB-RCS systems, the described scheme is instantiated by having the traffic management module simply classify traffic, e.g. into EF, AF and BE priorities, and by having data be enqueued in possibly separate buffers. The state of the buffers drives future capacity requests, while the encapsulator draws datagrams based on currently available DAMA resources. Should it not be possible to send an IP datagram completely within the available capacity, the spare fragments are processed in subsequent superframes.

On the other hand, the normative document [i.1] foresees the possibility to map PPDUs to different ACs, enabling the additional degree of freedom of splitting traffic between DA-AC and RA-AC. In this perspective, traffic sent through random access procedures is likely to experience lower delay by avoiding the capacity request process of DAMA, while experiencing good success rates by virtue of advanced MAC schemes such as CRDSA. Conversely, the terminal access rate limitation and the load control algorithm provide maximum boundaries for RA channels that suggest their usage for short bursts of data. These remarks hint a first possible approach for the integration of RA- and DA-ACs, relying on the former to send small or urgent datagrams as well as signalling, while reserving the latter for more predictable and less delay-sensitive traffic. Such a solution can also be flanked by a more dynamic strategy, as data could be split among channels taking into account current conditions in terms of load and congestion. In such an approach, RA data could also be opportunistically reallocated to DA slots if exclusive resources exceeding the current amount of DAMA datagrams are available.

A high level description of the discussed integration strategies is provided in Figure 6.24. The remainder of this section, instead, describes in greater details different solutions that stem from these core ideas, specifying the required modifications to the basic scheme of Figure 6.23 as well as discussing possible advantages and drawbacks.

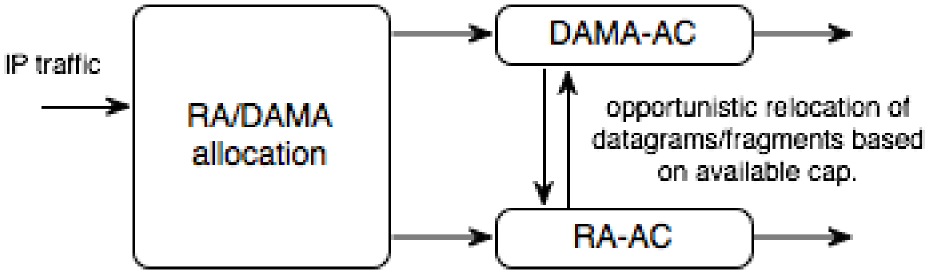

IP traffic
RA/DAMA allocation
DAMA-AC
RA-AC
opportunistic relocation of datagrams/fragments based on available cap.

Figure 6.24: Conceptual approach for RA- and DA-AC integration

6.8.2.1 Design requirements

An algorithm for the integration of RA and DA-AC should fulfil the following requirements:

1) The algorithm should take higher layer requirements into account (e.g. QoS and application requirements, such as avoiding out-of-order delivery at transport layer or large variations in delay variation).

2) The algorithm should take lower layer requirements into account (e.g. the size of L3 packets to limit the number of L2 fragments and the total L3 packet error rate).

3) The algorithm should take the current load of the DA channel and the RA channel into consideration.

4) The algorithm should take the access rate limitation mechanism on the RA channel into consideration.

5) The algorithm should allow L3 packets, which were originally intended for the RA-AC to be opportunistically relocated to the DA-AC, in case DA capacity is still available after serving all DAMA queues (i.e. DA-AC excess capacity).

6) The algorithm should not allow L3 packets that were originally intended for the DA-AC being opportunistically relocated to the RA-AC, because this would reduce the accuracy and consistency of the DAMA rate and queue measurements, which are used for the capacity request generation. Furthermore, it would counteract the first requirement since such a relocation could cause unacceptably delay variations or unacceptably deep misordering of L4 segments.

7) The algorithm should be able to improve the overall delay while satisfying the aforementioned requirements.

6.8.2.2 Example algorithm

Taking into account the features specified in clause 6.8.1.1, we propose in this section a possible algorithm for the integration of RA and DA-ACs. As discussed, higher layer requirements should be considered. In the first place, some higher layer applications may result sensitive to out of order delivery of L3 IP datagrams and lose in performance (e.g. TCP). Other issues may affect applications which are sensitive to high delay jitter (e.g. issues with the playback buffer of VoIP). Problems of out of order delivery could typically result if data is sent jointly over DA-AC and RA-AC with opportunistic relocation in both directions. Problems of higher delay jitter could typically result if data is sent over the RA-AC or jointly over DA-AC and RA-AC with opportunistic relocation in both directions. In particular, for the joint use of DA-AC and RA-AC a higher variation of delay jitter might result due to the different delays of DA and RA and data might arrive not in order either due to the different DA and RA delays (queuing and channel access delays) or due to the order of decoding RA and DA sections of a superframe.

For this reason, a first classification is applied to L3 PDUs to decide which packets should go over the DA-AC (i.e. the packets affected by the aforementioned issues) and which packets are eligible to go either over DA-AC or over RA-AC. Since this classification is based only on requirements coming from the higher layers, it is denoted as Higher-Layer-Classification (HLC).

Figure 6.25 shows this block at the left and also all the other blocks discussed in the following.

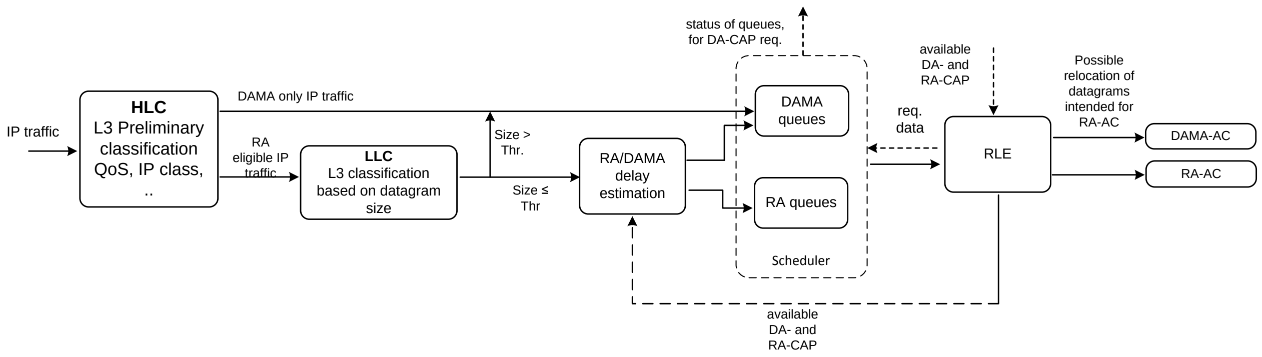


Figure 6.25: Proposed RA-DAMA integration strategy

Among all the packets that are tagged as eligible to be sent over the RA-AC or the DA-AC, a second classification is applied, which takes limitations coming from the lower layers (Lower Layer Classification, LLC) into account. Here, the limitation is basically the size of the L3 PDU. The justification for this is that large L3 PDUs will result in a large number of L2 PPDUs in the fragmentation process. If such a large number of L2 PPDUs is sent over the RA-AC (which naturally has a higher loss rate than the DA-AC) the probability of losing the L3 PDU increases. For this reason, a threshold is applied, so that packets above a given size are tagged to be sent only over the DA-AC, whereas the others remain eligible for RA-AC transmission. The value of the classification threshold depends on the tolerable packet loss rate as well as on the employed contention scheme on the RA-AC, i.e. Slotted Aloha (SA) or CRDSA.

Once the set of packets that may be sent over RA-AC satisfying HLC and LLC has been determined, a final choice is made to identify which of them should actually be enqueued in RA buffers. This decision aims at reducing the experienced delay. Therefore, the algorithm estimates the expected delay that a packet would undergo if sent over the DA-AC (i.e. taking into consideration the current DA load and queue state) and compares it to an estimation of the expected delay that the same packet would undergo if sent over the RA-AC (taking into consideration the current RA load and queue state as well as and restrictions given by the load control and the terminal access rate limitation settings).

If the expected delay over the RA-AC is lower than the one over the DA-AC, the packet is enqueued in the RA queue. If the expected delay for the RA-AC is equal or bigger to the DA-AC, then the packet is assigned to the DA-AC. The RLE encapsulator receives information about the DAMA capacity that the terminal was assigned in the current superframe. As long as DAMA capacity is available, it triggers the scheduler to send further L3 DAMA packets. Should the DAMA queues be empty but still capacity available, then an opportunistic relocation of packets from the RA queues towards the DAMA queues goes into effect. This is done to utilize the DAMA capacity assignments and lower the load and traffic over the RA-AC.

The methods for estimation of the RA and DAMA delay are explained in more detail clause 6.8.2.2.1.

6.8.2.2.1 DA-AC delay estimation

The only information which a terminal has available to estimate what will happen in the future is represented by the capacity requests it sends to the gateway. With this and the knowledge of the propagation delays, the terminal can estimate when the assignment should arrive in the future. The terminal furthermore has to assume that it will receive the capacity that it has requested, since it has no information about the overall system load status, which could indicate that a request may not be served fully due to overload. On the other hand, assuming that the requested capacity will be also assigned later on holds in low to average network loading conditions and may result in an overestimation of the serving rate only in situations where the network is getting saturated and overloaded. Such an overload should happen only seldom and not in nominal conditions if the network is properly dimensioned.

The approach for the DA-AC delay estimation is then the following:

a) Track all stated capacity requests within the terminal and keep book of them.

b) Compute for every future SF the capacity that the terminal expects to get assigned from the table of capacity requests it has sent earlier and considering the propagation delay until the assignment is expected.

c) Compute the number of bytes waiting in the queue (L_{DA}) in front of the current L3 PDU (L_{L3PDU}).

d) Integrate the expected capacity assignments from b) until the total number of bytes ($L_{DA} + L_{L3PDU}$) has been reached. The end time of the superframe in which the integration exceeds this value is then the expected delay (excluding the propagation delay which is a constant adding on top) of the current L3 PDU.

Table 6.13 shows an example of a possible Capacity Request storage table in a terminal.

Table 6.13: Example for capacity request storage table in the terminal

Type	Start	End	Capacity
CRA	T_1	∞	C_1 (Bytes/SF)
RBDC1	T_2	T_3	C_2 (Bytes/SF)
RBDC2	T_4	T_5	C_3 (Bytes/SF)
VBDC	T_3	T_3	C_4 (Bytes/SF)
VBDC	T_6	T_6	C_4 (Bytes/SF)

In Table 6.13, a Constant Rate Assignment (CRA) starting at T_1 is noted. As is the nature of CRA, the assignment is constant without a defined end. In the case of an update the CRA value in Table 6.13 needs to be updated as well. If CRAs are expressed in a byte rate per second (instead of number of bytes per SF), then a normalization to the SF duration needs to be done:

$$C_1 = CRA_{rate} \cdot T_{SF}$$

In Table 6.13, the two RBDCs are sent at different times. The first one requests C_2 Bytes per SF (if expressed in as byterate, a normalization to the SF duration as shown for the CRA needs to be done) in the time period starting at T_2 and lasting until T_3. The second RBDC requests an assignment of C_3 Bytes/SF in the time interval T_4 until T_5 .Finally, two VBDCs are sent to get granted in the SFs starting at time T_3 and T_6, both asking for a volume of C_4 Bytes.

Figure 6.26 illustrates these requests over time and shows the resulting expected capacity assignment C_{total} for every superframe, provided that all capacity requests are fully granted by the NCC without delay.

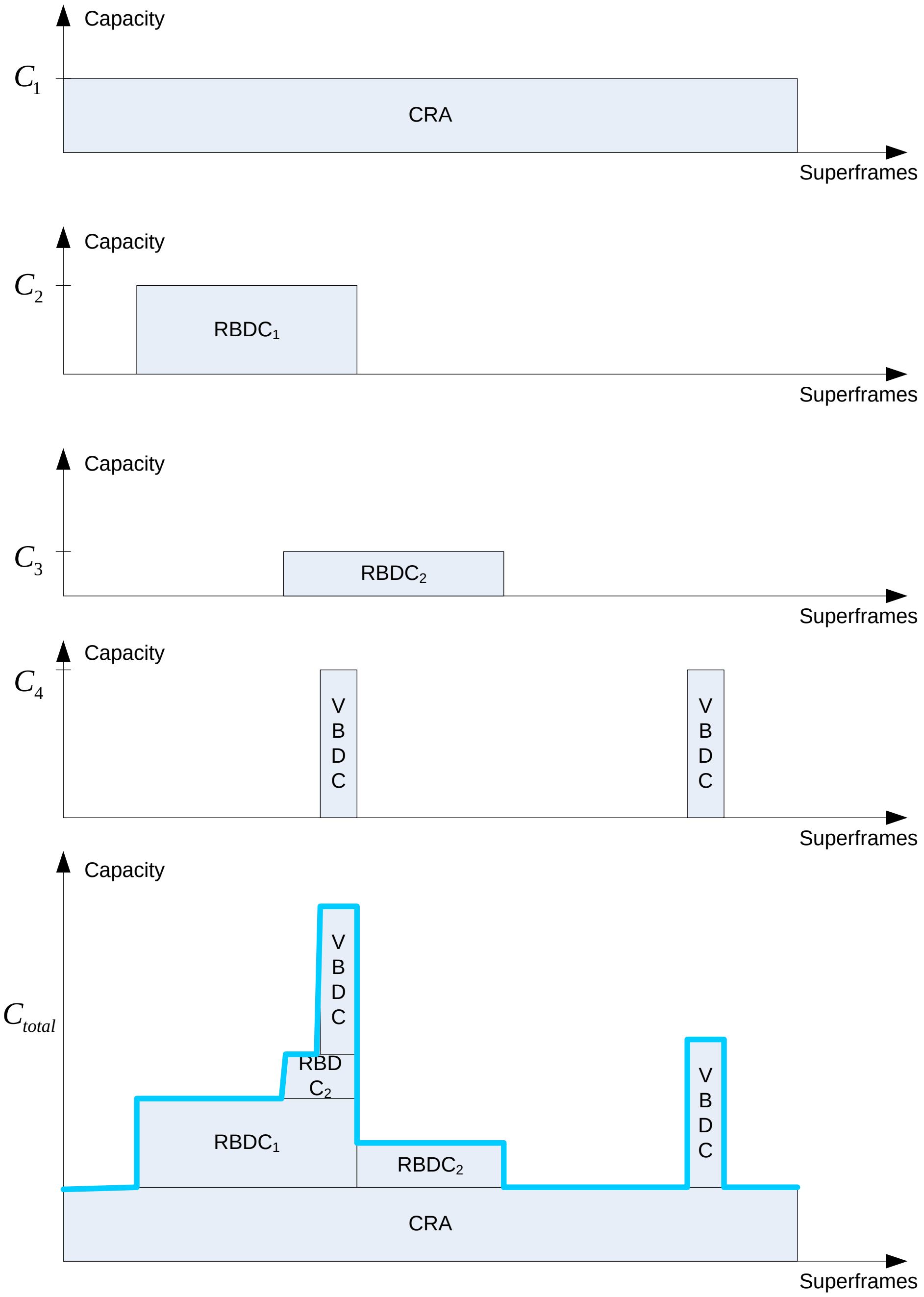


Figure 6.26: Example of CapReqs vs. time

Once the integral over C_{total} in Figure 6.26 exceeds the number of bytes stored in the queue + the current L3 PDU size, the SF in which the current L3 PDU is expected to be transmitted is found, and the expected queuing delay can be computed with the number of required SFs and the superframe duration.

$$\Delta_{DA} = N_{SF} \cdot T_{SF}$$

6.8.2.2.2 RA delay estimation

The serving rate is estimated from the signalled load control and access rate limitation values directly. Let:

- N_{max} = maximum number of fragments that can be sent per RA block.
- L_{max} = maximum number of successive RA blocks that can be accessed.

- p_b = backoff probability.
- I = number of idle RA blocks that the node should not access when rate access limitations have been reached.
- T_block = the duration of a RA block.

Ignoring backoff limitations, the maximum average rate at which fragments (i.e. unique payloads) can be transmitted can be estimated as:

$$\mu = \frac{N_{\max} L_{\max}}{(L_{\max} + I)T_{block}}$$

From this, the average number of fragments per transmission opportunity that can be sent considering backoff, i.e. the predicted service byterate, can be computed as:

$$R\wedge = (1-p_b)\mu$$

This value of $R\wedge$ is used to predict the delay undergone by a datagram if enqueued in a RA-AC buffer as

$$T_Q^{\wedge}(t) = \frac{F}{R_b^{\wedge}} + \sum_{i=1}^{n_p} \frac{F_i}{R_b^{\wedge}},$$

where

n_p = number of datagrams currently enqueued in buffer Q.

F = number of fragments of which the datagram to be enqueued is composed.

F_i = number of fragments of which datagram i is composed.

6.8.3 Integration of DA-AC and RA-AC for user data and signalling transmission

The RA-DAMA integration strategy described in the previous section focuses on integrating the transmission of user data, while the signalling (e.g. capacity requests) is sent either in unsolicited dedicated control slots or in state-of-the art Slotted ALOHA (SA) channels.

The provision of a RA-AC for data transmission may have a positive effect on the overall delay. In this section, another interesting application case for a RA-AC is investigated, namely to replace the SA minislot signalling carrier with a more efficient CRDSA RA-AC which may also be utilized for user data transmissions.

Doing this brings two main benefits:

1) The spectral efficiency of the signalling carrier increases compared to SA, allowing more signalling to be carried in the same amount of bandwidth.

2) The CRDSA signalling carrier can be also used for additional user data transmission in addition to the signalling which may further increase the efficiency.

In the following, we concentrate on the case of having signalling sent over a RA-AC (referred to as RA-SIG-AC) together with additional data traffic.

From a practical implementation point of view, the fact that the signalling messages are L2 packets while data are L3 packets needs to be taken into account. This means that the L2 Signalling and the L3 IP packets cannot be enqueued in the same buffer but need separate ones.

Figure 6.27 illustrates the integration of the L3 IP data with the L2 signalling data over a common RA-SIG AC.

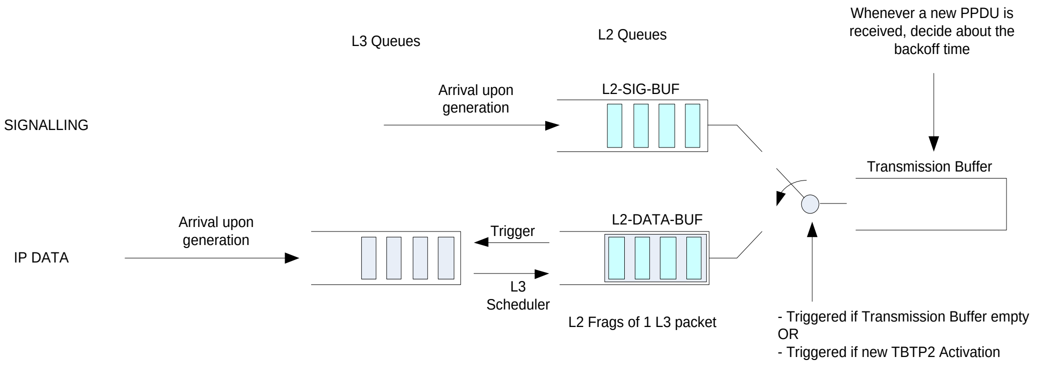


Figure 6.27: Integration concept of IP data and L2 signalling over a common RA-SIG allocation channel

Since the signalling messages are L2 messages, there is no L3 queue existing, but signalling messages are directly enqueued in a L2 queue at the encapsulator upon their arrival.

The L3 IP data on the other hand are enqueued in a L3 buffer. The encapsulator triggers the L3 scheduling of a new L3 packet whenever there is remaining space in the RA-SIG-AC. Once the scheduled L3 packet arrives, its fragments are enqueued in a separate L2 buffer. This is done since it is considered meaningful to give signalling messages priority over user data packets, since blocking signalling messages will result in delayed or missing DAMA allocations, which is not desirable and has a negative impact on the DAMA performance. Keeping separate queues and scheduling them with a priority scheduler is foreseen to ensure the priority of scheduling over data messages.

It should be noted that only the L2 fragments of one L3 packet may reside in the L2-DATA-BUF queue, while in the L2-SIG-BUF queue several different signalling messages may be stored.

The L2 PPDU which is selected for transmission by the L2 scheduler is then stored in the transmission buffer. There is only a single transmission buffer for the RA-SIG-AC. A new L2 scheduling round only starts if the current transmission buffer is empty.

Figure 6.28 illustrates the procedure to be followed whenever a new L2 signalling message arrives.

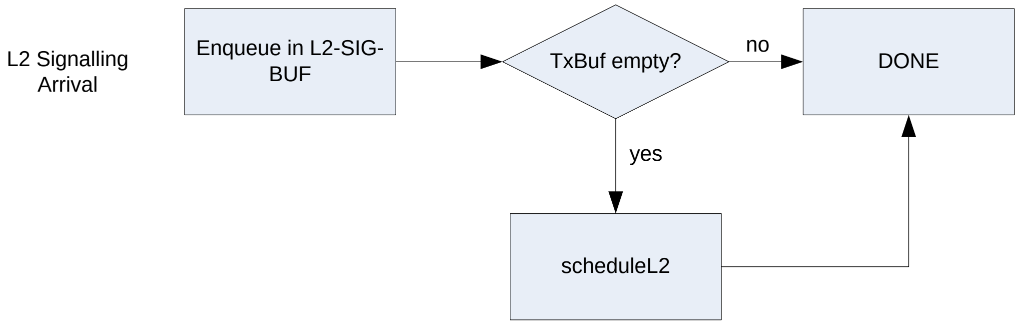


Figure 6.28: Flowchart for actions upon arrival of a L2 signalling message

As explained before, the L2 signalling message is in any case enqueued into the L2-SIG-BUF. In case the Transmission Buffer (TxBuf) is empty at this time, a new L2 scheduling round is initiated immediately. If there is still data pending in the TxBuf, then the procedure finishes.

The procedure upon arrival of a L3 data packet, scheduled by the L3 scheduler is the same (see Figure 6.29), except that the IP data are enqueued in a separate buffer called L2-DATA-BUF in Figure 6.27.

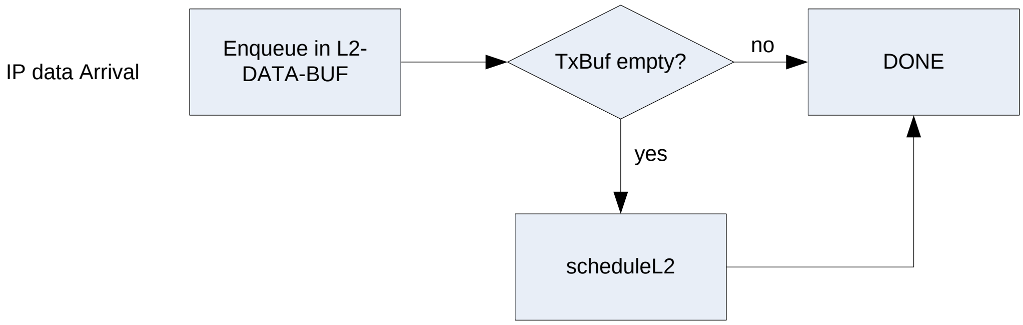


Figure 6.29: Flowchart for actions upon arrival of a L2 signalling message

Transmission over the RA-SIG-AC can be triggered by two events:

- A SF activation event: Occurs at the start of a new SF.
- A L2 scheduling event: Occurs whenever the L2 scheduler is triggered.

Figure 6.30 illustrates the flow chart upon a SF-activation event.

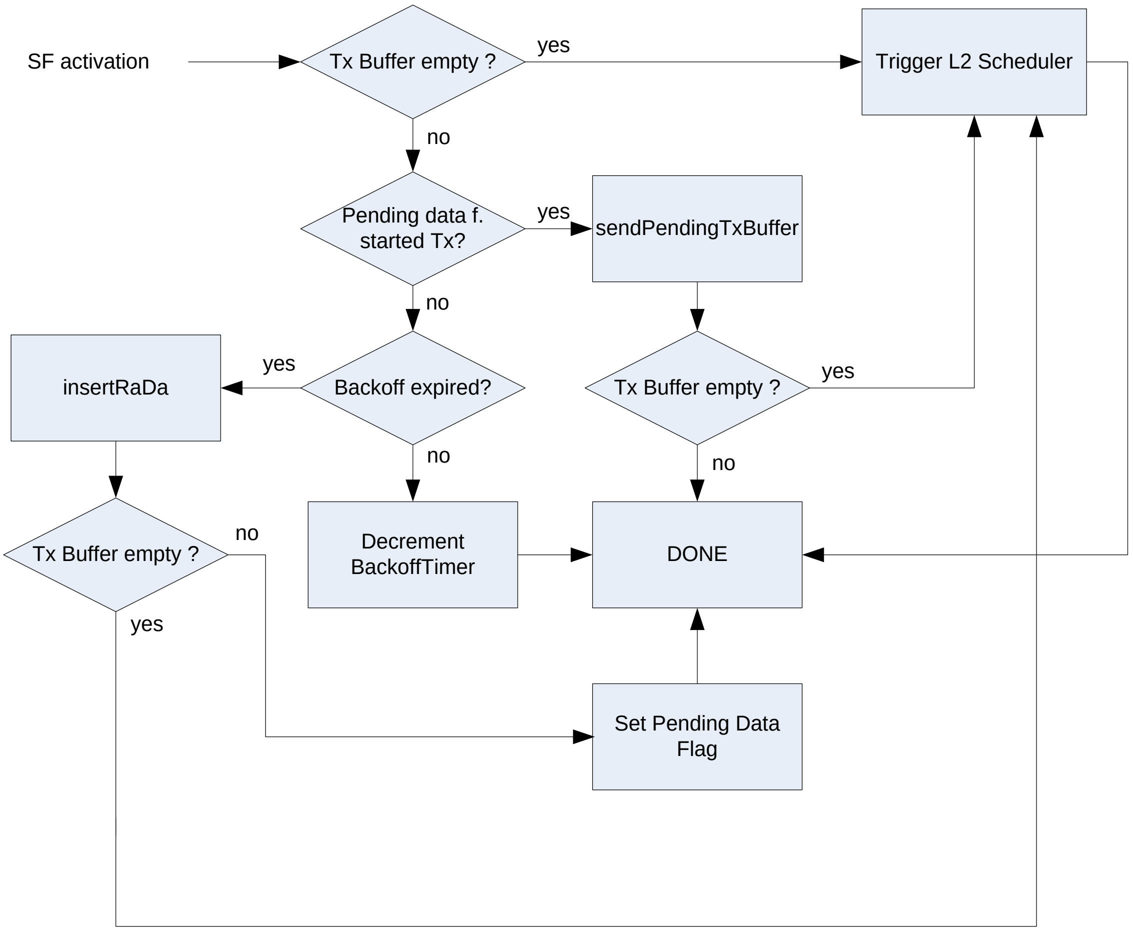


Figure 6.30: Flow chart upon a SF activation event

Whenever the time of activation of a new SF has arrived, this procedure is executed. First the status of the TxBuf is checked. In case the TxBuf is currently empty, a new round of the L2 scheduler is triggered and the procedure is finished:

- If there is data in the transmission buffer, it needs to be checked whether the fragments in the buffer are part of an already initiated transmission of an existing higher-layer PDU (i.e. pending data) or whether they are the fragments of a newly arrived higher-layer PDU. This distinction is necessary to comply with the RA load control specification for the following reason: Set the case that the L2 scheduler delivered a new L2 message in the TxBuf. According to section 9.7.3.2 in the normative document [i.1], at this time a decision about the backoff needs to be done. According to the normative document [i.1], the decision can only be made if the data is already received, i.e. located in the TxBuf, not before. In case a backoff was decided, then the SF-activation procedure needs to check the expiry of the backoff time.

- Set another case where data is in the TxBuf, but this data is part of a transmission initiated already earlier but which could not yet be finished. In this case, the expiration of the backoff counter is not relevant, but a decision about avoiding the usage of the current transmission probability needs to be made.

In order to differentiate between the two cases, a flag for indication of pending data (Pending Data flag) is introduced. Other ways of implementing the distinction are possible as long as the procedure complies with the load control specification.

In case the data payload currently in the TxBuf is pending data of an earlier transmission, the sendPendingTxBuffer subprocedure then checks whether the RA transmission opportunity needs to be omitted or not, i.e. if the maximum number of consecutive blocks is not yet exceeded, the maximum number of unique payloads per block is not yet exceeded and the random decision for avoiding the transmission opportunity was in favour of transmission, then the payload is sent.

In case the full payload could be sent, the TxBuf will be empty. In this case another L2 scheduling round is triggered to exploit the full capacity of the RA-AC. If data is still in the TxBuf then the load control limitations have been reached and the current assignment round is over.

In case the data in the TxBuf is a new payload, then the procedure checks whether the backoff time has expired or not. In case it has not yet expired, the BackoffTimer is decremented and the current allocation round is finished.

In case the backoff has expired, the new payload is sent over the RA-AC (insertRaDa block), where again the check of the load control limitations as for the sendPendingTxBuffer is applied.

In case the payload could be sent completely, then also here another L2 scheduling is initiated. If the payload could not be sent completely, the pending data flag is set and the current allocation round is over.

The procedure of a new L2 scheduling round is depicted in Figure 6.31.

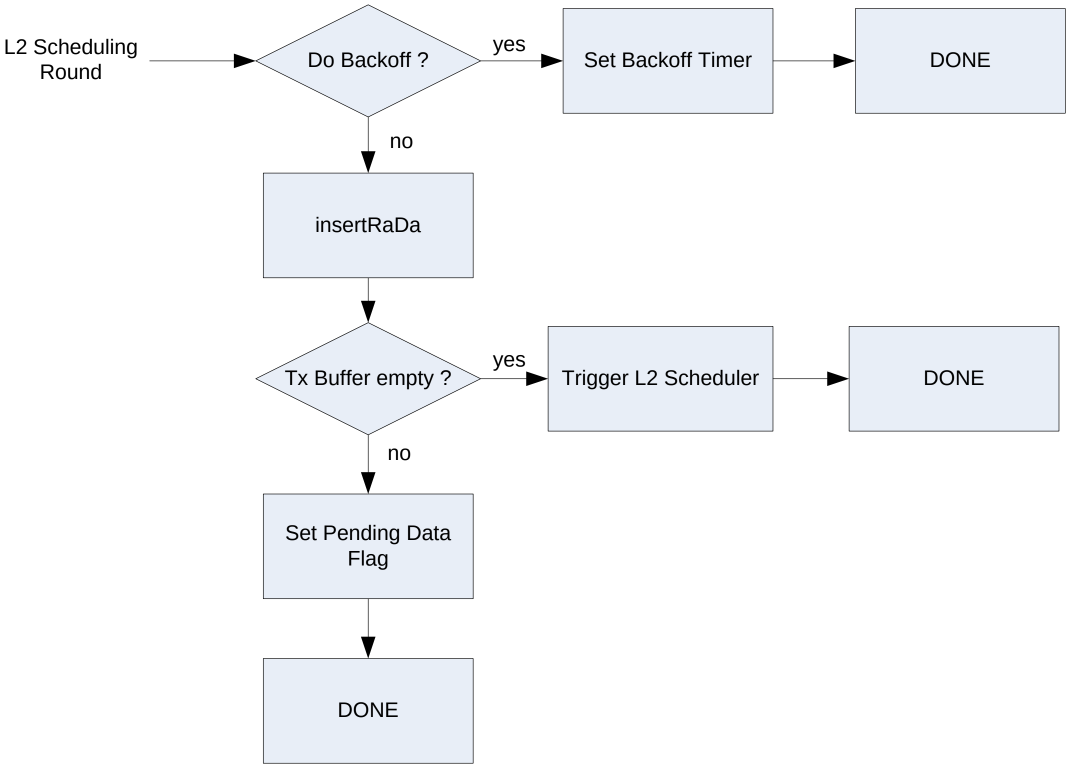


Figure 6.31: Flowchart of the procedure upon a new L2 scheduling round

For the L2 scheduler, different scheduling strategies can be implemented. Here we propose a strict priority scheduler which always assigns priority to signalling packets for the aforementioned reasons. In other words, whenever data is waiting in the L2-Sig-Buf and the L2-Data-Buf, the signalling messages will always be scheduled first.

After scheduling a message (signalling or data) and enqueuing in the TxBuf the decision of the backoff is made. In case the transmission should be backoffed (with probability p_b), the backoff time is set to the back_off_time specified in the received signalling fields and the scheduling round is over.

In case of no backoff, the insertRaDa function checks whether the load control limitations (max number of consecutive payloads, max number of unique payloads per block) are exceeded and in case transmits the payload over the RA-AC.

If some data is still remaining in the buffer, the pending data flag is set and the scheduling round is over.

If everything could be sent, another L2 scheduling round is initiated (recursively) and the current allocation round is over.

7 M&C Functions Supported by L2S

7.1 Control of EIRP

The manufacturer of the RCST needs to state precisely the operating range of the EIRP control of the RCST. This is because different system operators may use different strategies for RCST EIRP control. In some system designs, a wide rain fade margin may be expected and tight RCST uplink power control exercised. In other system designs, RCST uplink power control may not even be used depending on system cost trade-offs.

RCST EIRP control may be exercised by the RCST itself or by the NCC.

It is generally anticipated that in most system designs the RCST EIRP will be adjusted up or down in nominal 0,5 dB increments over the operating EIRP range of the RCST by commands from the NCC. This will be in response to direct or indirect measurements of the link margin of the RCST in question.

For large step changes in EIRP level it is unreasonable to expect the RCST to provide a nominal 0,5 dB accuracy and so in this case the specification only calls for the resulting power change to be within 20 % of the dB value of the requested step change.

In this circumstance it is expected that the EIRP step change will be followed by incremental up or down EIRP changes in nominal 0,5 dB steps.

Whenever the EIRP of the terminal is increased, there is a possibility for spectral regrowth (for linear modulation) to occur, impacting the performance of the adjacent channels. There is a number of different approaches to solve this problem. These are left to the system designer. One possibility is that the OBO could be controlled in conjunction with the NCC. The NCC determines from time to time the operating point on the output power versus input power curve of the HPA. This can be realized for example by requesting RCST Transmit Power headroom as described below.

7.1.1 RCST Transmit Power Headroom

This clause provides some considerations of the "power headroom" parameter, including why it is useful and how the value can be determined.

The "headroom" parameter reported by the RCST in the control PPDU (see Table 8-8 of [i.1]) indicates the difference (in dB) between the actual transmit power and the maximum possible. The main rationale for reporting this parameter is to aid the decision process in systems that employ countermeasure techniques against fading or other channel variations.

The most reliable signal quality indicator at the NCC is typically the *C/N* of the received bursts. However, if the back-off of the RCST is not known, the NCC cannot easily determine the RCST's current capability. Schemes that attempt to track the back-off by accumulating the commanded power variation can easily get out of step, if messages are missed or if saturation occurs in places other than expected. If the headroom *H* is known explicitly, the capability of the RCST can be expressed as:

$$\left.\frac{C}{N_0}\right|_{\max} = \frac{C}{N} + 10\log_{10}\left(R_S\right) + H$$

The RCST will be able to use any return link transmission mode that has a threshold *C/N0* not exceeding $C/N0|_{\max}$. This is particularly an issue at times where the channel conditions are improving, such as at the tail end of a fade event: If the headroom is not accounted for in these situations, there is a risk that the RCST will continue simply to back-off the power. If this happens, the *C/N* never improves and the RCST may thus never switch to a higher-rate and/or more efficient mode of transmission. This is generally less of an issue while conditions are worsening; power control will tend to maximize the power so the headroom is usually small at that end of the process. These considerations are what led to inclusion of headroom reporting in the RCS2 return link signalling.

The headroom can be determined at the RCST in a number of different ways, with different accuracy and cost implications. In the simplest form, the headroom can be estimated from the IF attenuator setting. This requires no additional hardware; in particular, it does not require any communication between the modem part and the power amplifier proper. However, this method does not allow for the compression in the power amplifier, so it will tend to be less accurate when operating close to maximum power.

The method can be improved for example by using a look-up table that models the compression characteristic. However, in order to bring a meaningful improvement, such a table really needs to be calibrated for individual power amplifiers - the production spread of characteristics can be quite considerable. This complicates the installation, or at least forces the pairing of modem (indoor unit, IDU) and power amplifier to be made at the factory so they can be calibrated together. This may be acceptable in many situations, but gives complications later if either of the units needs to be replaced.

In any case, the look-up table method does not account for changes due to temperature variations and aging of the power device, which will alter the relationship between the IF level and the position of the compression curve.

The most reliable method of determining the headroom is obviously to have a power detector as part of the RF amplifier unit. The power detector itself can be a very simple and cheap circuit; complications and cost arise mainly from the need to communicate the measurements to the modem/IDU. This is particularly the case for stand-alone power amplifiers (BUC's). These typically have a single-cable connection that carriers DC power, IF signal and a reference tone for the up-converter. Adding the capability to signal back from the BUC to the IDU, for example using DiseqC, necessitates bulky and expensive diplexers as well as other additional hardware. This is the main reason explicit power detection is not commonly used in low-cost units.

It is easier to accommodate the power measurement signalling in more integrated RF units, for example those that combine transmit and receive functions ("transceivers"). It is typically much easier and cheaper to accommodate the signalling on the receive cable, which does not carry high levels of power. Such more sophisticated units may already have a signalling path for other monitoring functions.

Fully-integrated terminals, in which the modem and RF devices are in the same enclosure, of course have no such issues with reporting of the measured power.

7.2 Pointing Alignment Support

7.2.1 Point Alignment Support descriptor

The descriptor contains a single data structure the contents of which depends on its type, which is indicated in the alignment_control_type field. The alignment_control_type field values are listed in Table 7.1. Depending on this field value, the descriptor may be encapsulated in TIM-B or TIM-U signalling tables. The NCC may encapsulate multiple pointing alignment support descriptors in the same TIM-U or TIM-B signalling tables.

Table 7.1: Alignment control types

Bit	Usage	Occurrence
0	Broadcasted declaration	TIM-B
1	Broadcasted NOC reference	TIM-B
2-63	Reserved	
64	Alignment procedure with use of a specific POPID	TIM-U
65	Alignment procedure w/o changing POPID	TIM-U
66	Burst based alignment	TIM-U
67	CW based alignment, dynamic EIRP	TIM-U
68	CW based alignment, fixed EIRP	TIM-U
69-95	Reserved	
96	Alignment feedback	TIM-U
97-127	Reserved	
128-255	User Defined	

With alignment control type of "0", the NCC may broadcast whether or not pointing alignment is required in the network, whether or not the NCC supports CW-based return link alignment, whether or not the NCC supports IB-based return link alignment, and the forward link SNR threshold that should be exceeded to complete forward link alignment procedure. With this alignment control type, the NCC may also convey information pertinent to a user-defined alignment procedure.

With alignment control type of "1", the NCC may broadcast an ASCII string that provides a reference (e.g. a phone number) to a Network Operations Centre, which may provide assistance with the return link alignment procedure.

With alignment control type of "65", the NCC may unicast to a RCST the remaining duration in which the RCST may continue with return link alignment procedure and the threshold values for co-polar and cross-polar measurements. Upon reception of this descriptor, the RCST resets its remaining duration timer to the number of seconds indicated in the remaining duration field of this descriptor.

With alignment control type of "64", the NCC may unicast to a RCST an alignment population ID value that is different from the RCST's operational population ID in addition to the information that can be conveyed with the alignment control type of "65". Upon receiving this descriptor, the RCST tunes to the forward link signalling service identified by the alignment population ID in the RMT table.

With alignment control type of "66", the NCC may unicast to a RCST a 16-bit pattern that the RCST transmits in installation bursts during the IB-based return link alignment. DVB-RCS2 dictates using the Logon SDUs in dedicated access logon slots allocated by the NCC to transmit installation bursts. RCST replicates the 16-bit pattern as many times as necessary to fill in the payload of the Logon SDU for installation burst transmission. Section-3.1.2.1 elaborates further on the use of Logon SDUs for installation burst transmission. The logon slots dedicated for installation burst transmission are allocated by the NCC in the BTP.

With alignment control type of "68", the NCC may unicast to a RCST the start time, the duration, and the frequency values that the RCST should use during the CW-based return link alignment procedure.

With alignment control type of "67", the NCC may unicast to a RCST the EIRP that the RCST should use during the CW-based return link alignment procedure in addition the information that can be conveyed with the alignment control type of "68". Note that devices that cannot dynamically control their EIRP should terminate the return link alignment procedure with failure upon receiving this descriptor. The RCST may provide visual aids to the human operator indicating the reason for failure.

With alignment control type of "96", the NCC may unicast to a RCST the CNR, co-polar, and cross-polar measurements read on the return link transmission from the RCST. In addition, the NCC may convey alignment status of the RCST, which may be "in-progress", "failure", and "success". The RCST should continue the return link alignment procedure for as long as its alignment status is "in-progress" provided that the most recent remaining duration value from the NCC has not been exceeded.

7.2.2 Elements in Logon SDU and Logon Response Descriptor

7.2.2.1 Logon SDU

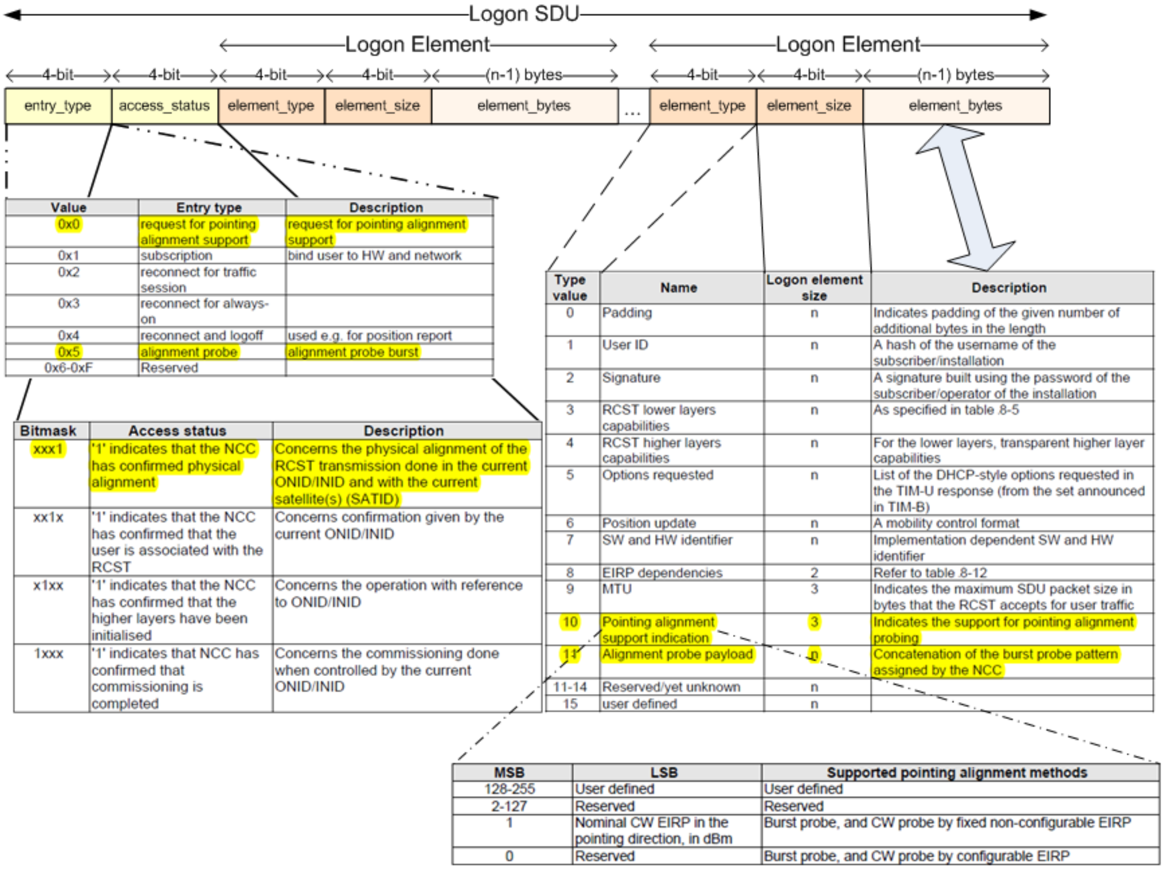
The Logon SDU contains a 4-bit entry_type and a 4-bit access_status fields.

The RCST may request for alignment support by assigning 0x00 to the entry_type field in the Logon SDU. When the Logon SDU is used to transmit an installation burst during the return link alignment, the RCST assigns 0x05 to the `entry_type`.

If the least significant bit in the access_status field is '1', the RCST indicates that the NCC has confirmed earlier that the RCST has completed alignment procedure. The NCC may disregard this indication and enforce the RCST to repeat the alignment procedure. When the Logon SDU is used to transmit an installation burst during the return link alignment, the RCST assigns 0x00 to the access_status.

The Logon SDU may contain a number of Logon Elements after the entry-type and access_status fields. Each Logon Element may be of variable size. Each Logon Element contains a 4-bit type and a 4-bit size fields. Logon element of type '10' contains RCST pointing alignment support capabilities. This is a 3-byte logon element that indicates the nominal EIRP in the pointing direction if the RCST is a fixed-EIRP device. The logon element may also be used to indicate that the RCST supports CW-based, IB-based, and dynamic-EIRP return link alignment procedures. When the Logon SDU is used to transmit an installation burst during the return link alignment, it only contains logon elements of type '11'. In this case, the Logon SDU encapsulates as many logon elements as necessary to fill in the space in the logon slot. If the remaining space is only 1 byte, then the RCST fills in this space with the most significant byte of the 16-bit pattern indicated in the Pointing Alignment Support descriptor from the NCC.

Figure 7.1 summarizes the Logon SDU with highlights over the fields regarding the pointing alignment support.



Value	Entry type	Description
0x0	request for pointing alignment support	request for pointing alignment support
0x1	subscription	bind user to HW and network
0x2	reconnect for traffic session	
0x3	reconnect for always-on	
0x4	reconnect and logoff	used e.g. for position report
0x5	alignment probe	alignment probe burst
0x6-0xF	Reserved	

Bitmask	Access status	Description
xxx1	'1' indicates that the NCC has confirmed physical alignment	Concerns the physical alignment of the RCST transmission done in the current ONID/INID and with the current satellite(s) (SATID)
xx1x	'1' indicates that the NCC has confirmed that the user is associated with the RCST	Concerns confirmation given by the current ONID/INID
x1xx	'1' indicates that the NCC has confirmed that the higher layers have been initialised	Concerns the operation with reference to ONID/INID
1xxx	'1' indicates that NCC has confirmed that commissioning is completed	Concerns the commissioning done when controlled by the current ONID/INID

Type value	Name	Logon element size	Description
0	Padding	n	Indicates padding of the given number of additional bytes in the length
1	User ID	n	A hash of the username of the subscriber/installation
2	Signature	n	A signature built using the password of the subscriber/operator of the installation
3	RCST lower layers capabilities	n	As specified in table .8-5
4	RCST higher layers capabilities	n	For the lower layers, transparent higher layer capabilities
5	Options requested	n	List of the DHCP-style options requested in the TIM-U response (from the set announced in TIM-B)
6	Position update	n	A mobility control format
7	SW and HW identifier	n	Implementation dependent SW and HW identifier
8	EIRP dependencies	2	Refer to table .8-12
9	MTU	3	Indicates the maximum SDU packet size in bytes that the RCST accepts for user traffic
10	Pointing alignment support indication	3	Indicates the support for pointing alignment probing
11	Alignment probe payload	n	Concatenation of the burst probe pattern assigned by the NCC
11-14	Reserved/yet unknown	n	
15	user defined	n	

MSB	LSB	Supported pointing alignment methods
128-255	User defined	User defined
2-127	Reserved	Reserved
1	Nominal CW EIRP in the pointing direction, in dBm	Burst probe, and CW probe by fixed non-configurable EIRP
0	Reserved	Burst probe, and CW probe by configurable EIRP

Figure 7.1: Logon SDU elements for pointing alignment support

7.2.2.2 Logon Response descriptor

The Logon Response descriptor contains a 4-bit RCST_access_status field. This field has the same encoding with the `access_status` field in the Logon SDU. If the least significant bit of the RCST_access_status field is '1', then the NCC confirms in the Logon Response descriptor that the RCST does not need to repeat the pointing alignment procedure.

7.2.3 Example signalling exchanges

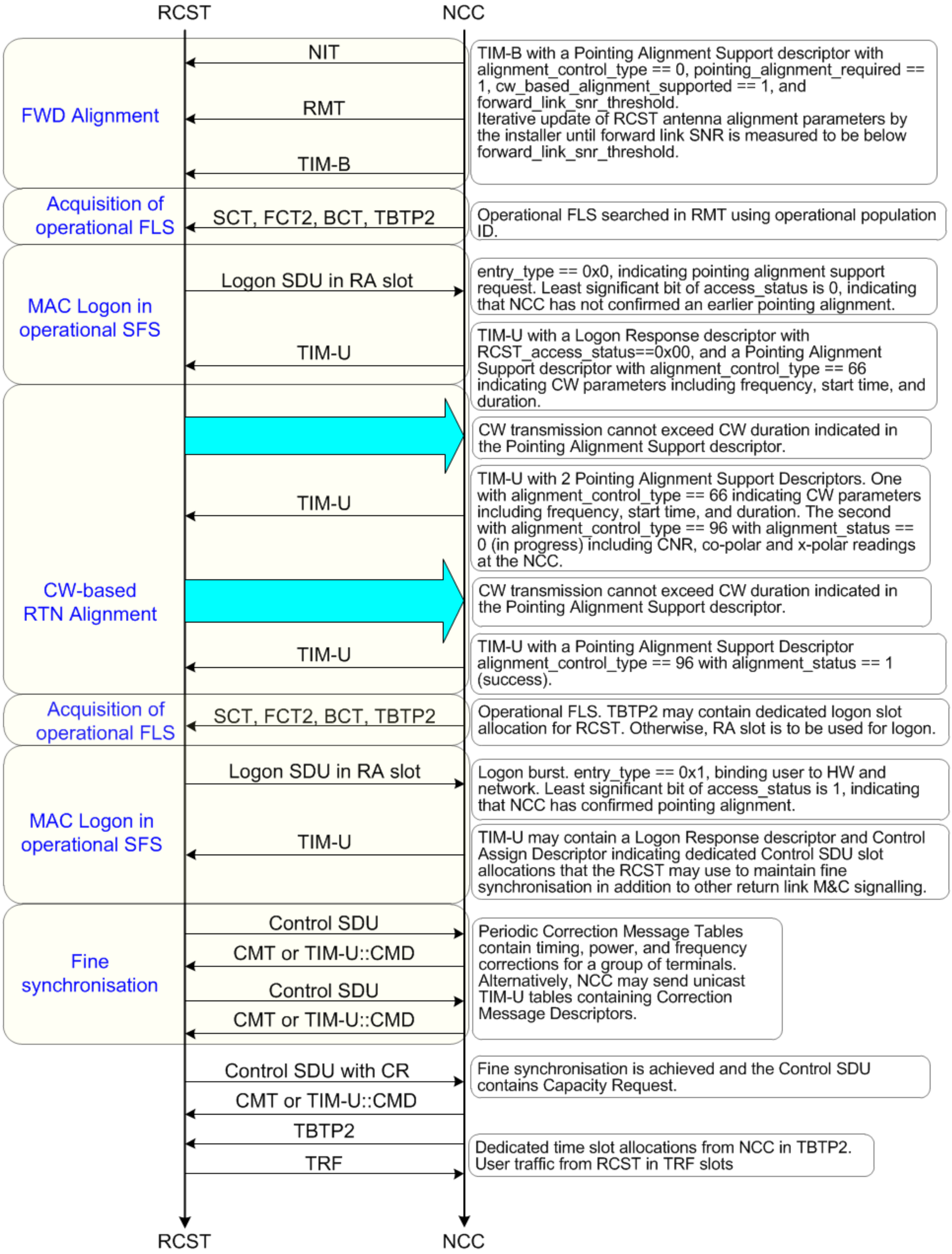
7.2.3.1 Continuous-wave (CW) alignment

Figure 7.2 shows an example CW-based return link alignment message exchange sequence between the RCST and the NCC. It is assumed that the RCST is a fixed-EIRP device, which should be indicated in the Logon SDU in the pointing alignment support indicator logon element.

Upon power-up, the RCST tunes to the start-up TDM to download periodical NIT, RMT, and TIM-B messages. The TIM-B messages from the NCC indicate that the NCC requires RCST pointing alignment support before joining in the network, that the NCC supports CW-based return link alignment, and the minimum forward link SNR threshold that should be exceeded before the return link alignment procedure may commence. The RCST installer adjusts the antenna alignment during forward link alignment before initiating the return link alignment procedure.

Upon completing the forward link alignment, the RCST finds and tunes to its FLS TDM stream corresponding to its population ID. Upon reception of SCT, FCT2, BCT, and TBTP2 tables, the RCST sends a Logon SDU in a RA Logon slot. The Logon SDU indicates that the RCST requests pointing alignment support and that the NCC has not confirmed an earlier alignment completion. In response, the NCC sends a TIM-U containing two descriptors. The first is a Logon Response descriptor with `RCST_access_status`==0x00. It is recommended that the NCC does not assign any RCS-MAC address to the RCST at this stage, because it is not certain yet that the return link alignment procedure will succeed. Note that lower layer unicast signalling on the forward link always uses the 48-bit RCST hardware identifier as the unicast address. The second descriptor in the TIM-U is a Pointing Alignment descriptor containing `alignment_control_type` of '66' indicating that the NCC supports CW-based return link alignment procedure. This is accompanied with the frequency, start time, and duration for the CW signal that the RCST needs to transmit. The NCC may send another Pointing Alignment descriptor before the CW transmission is over, effectively, extending the duration of the CW transmission (this is not the case in the example). In response to the CW transmission, the NCC sends a TIM-U with two Pointing Alignment descriptors. One of the descriptors provides feedback in terms of CNR, co-polar, and cross-polar readings. The second descriptor instructs additional CW transmission.

In response to the second CW transmission from the RCST, the NCC sends a TIM-U containing a Pointing Alignment descriptor that indicates return link alignment success. This is followed by the RCST accessing the RA channel to send a logon SDU. In response, the NCC encapsulates in a TIM-U a Logon Response descriptor and a Control Assign descriptor. The Logon Response descriptor assigns all necessary addresses to the RCST since the return link alignment has been achieved. The Control Assign descriptor allocates to the RCST a number of periodic Control SDU slots for achieving and maintaining fine synchronization in addition to other return link M&C signalling. The TIM-U may at this stage also contain descriptors necessary to fully configure the RCST.



TIM-B with a Pointing Alignment Support descriptor with alignment_control_type == 0, pointing_alignment_required == 1, cw_based_alignment_supported == 1, and forward_link_snr_threshold.
Iterative update of RCST antenna alignment parameters by the installer until forward link SNR is measured to be below forward_link_snr_threshold.

Operational FLS searched in RMT using operational population ID.

entry_type == 0x0, indicating pointing alignment support request. Least significant bit of access_status is 0, indicating that NCC has not confirmed an earlier pointing alignment.

TIM-U with a Logon Response descriptor with RCST_access_status==0x00, and a Pointing Alignment Support descriptor with alignment_control_type == 66 indicating CW parameters including frequency, start time, and duration.

CW transmission cannot exceed CW duration indicated in the Pointing Alignment Support descriptor.

TIM-U with 2 Pointing Alignment Support Descriptors. One with alignment_control_type == 66 indicating CW parameters including frequency, start time, and duration. The second with alignment_control_type == 96 with alignment_status == 0 (in progress) including CNR, co-polar and x-polar readings at the NCC.

CW transmission cannot exceed CW duration indicated in the Pointing Alignment Support descriptor.

TIM-U with a Pointing Alignment Support Descriptor alignment_control_type == 96 with alignment_status == 1 (success).

Operational FLS. TBTP2 may contain dedicated logon slot allocation for RCST. Otherwise, RA slot is to be used for logon.

Logon burst. entry_type == 0x1, binding user to HW and network. Least significant bit of access_status is 1, indicating that NCC has confirmed pointing alignment.

TIM-U may contain a Logon Response descriptor and Control Assign Descriptor indicating dedicated Control SDU slot allocations that the RCST may use to maintain fine synchronisation in addition to other return link M&C signalling.

Periodic Correction Message Tables contain timing, power, and frequency corrections for a group of terminals. Alternatively, NCC may send unicast TIM-U tables containing Correction Message Descriptors.

Fine synchronisation is achieved and the Control SDU contains Capacity Request.

Dedicated time slot allocations from NCC in TBTP2.
User traffic from RCST in TRF slots

Figure 7.2: Example message exchange with CW-based return link alignment, fixed-EIRP RCST, the same population ID for operational and alignment phases

7.2.3.2 Installation-burst (IB) alignment

Figure 7.3 shows an example IB-based return link alignment message exchange sequence between the RCST and the NCC. It is assumed that a different alignment population ID is maintained than the operational population ID.

Upon power-up, the RCST tunes to the start-up TDM to download periodical NIT, RMT, and TIM-B messages. The TIM-B messages from the NCC indicate that the NCC requires RCST pointing alignment support before joining in the network, that the NCC supports IB-based return link alignment, and the minimum forward link SNR threshold that should be exceeded before the return link alignment procedure may commence. The RCST installer adjusts the antenna alignment during forward link alignment before initiating the return link alignment procedure.

Upon completing the forward link alignment, the RCST finds and tunes to its FLS TDM stream corresponding to its population ID. Upon reception of SCT, FCT2, BCT, and TBTP2 tables, the RCST sends a Logon SDU in a RA Logon slot. The Logon SDU indicates that the RCST requests pointing alignment support and that the NCC has not confirmed an earlier alignment completion. In response, the NCC sends a TIM-U containing three descriptors. The first is a Logon Response descriptor with `RCST_access_status`==0x00. It is recommended that the NCC does not assign any RCS-MAC address to the RCST at this stage, because it is not certain yet that the return link alignment procedure will succeed. Note that lower layer unicast signalling on the forward link always uses the 48-bit RCST hardware identifier as the unicast address. The second descriptor in the TIM-U is a Pointing Alignment descriptor containing `alignment_control_type` of '64' indicating the use of an alignment population ID that is different from the RCST's operational population ID. The third descriptor is another Pointing Alignment Support descriptor with `alignment_control` type of '66' indicating the 16-bit pattern to be used in installation bursts. The RCST searches in the RMT the FLS TDM that corresponds to the alignment population ID and tunes to this TDM and acquires the FLS. The BTP (SCT/FCT2/BCT/TBTP2) in this FLS contains DA logon slot allocations for the RCST.

The RCST sends a Logon SDU with alignment probe (installation burst) in the DA logon slots allocated by the NCC. In response, the NCC sends a TIM-U with a Pointing Alignment descriptor providing feedback in terms of CNR, co-polar, and cross-polar readings. TIM-U also contains a Correction Message descriptor exploiting the return link alignment procedure also for fine synchronization achievement purposes. In parallel, the RCST receives additional DA logon slot allocation in TBTP2.

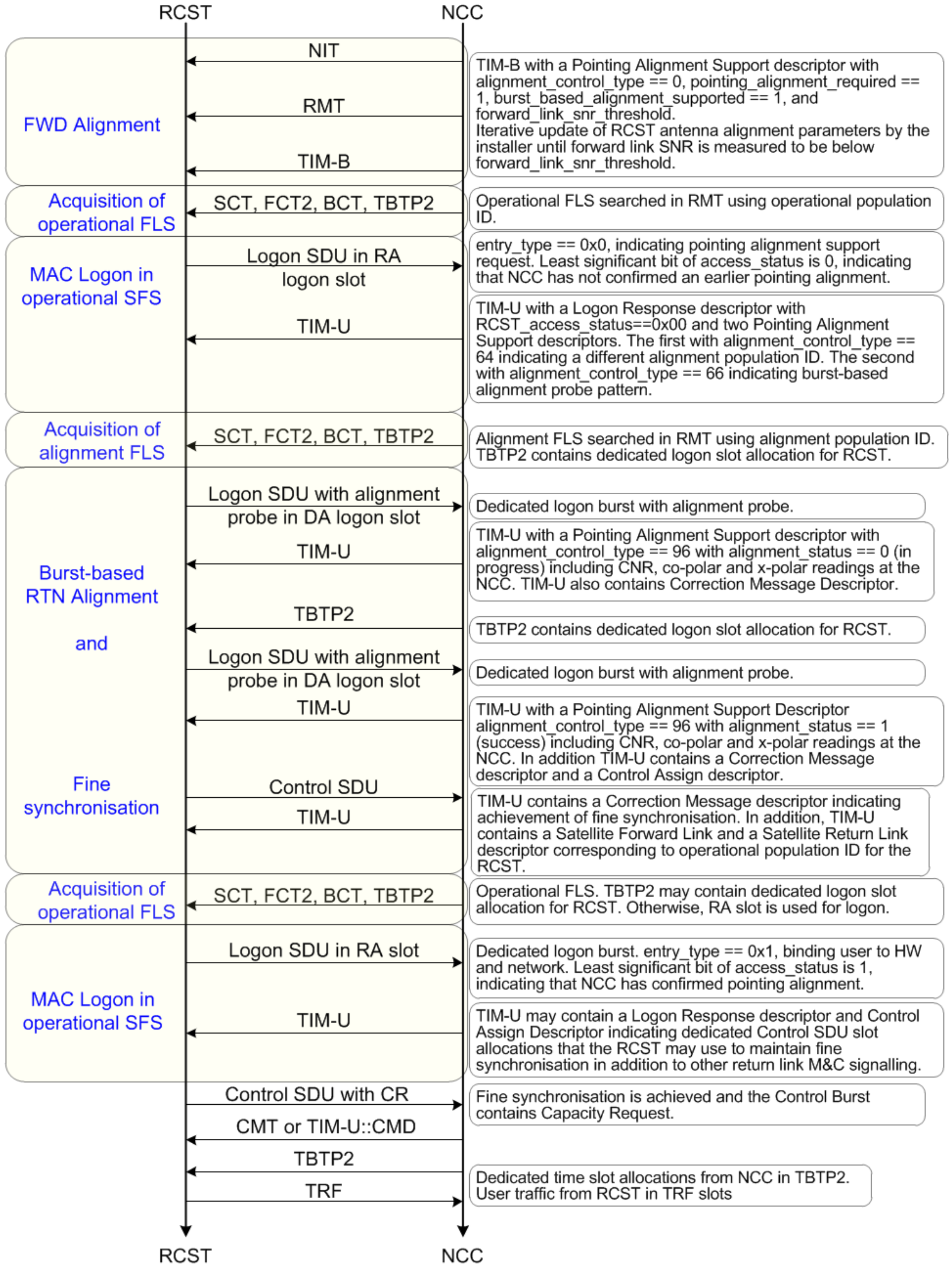
The RCST sends additional Logon SDU in the DA logon slots. In response, the NCC sends a TIM-U containing a Pointing Alignment descriptor that indicates return link alignment success. In addition, the TIM-U contains a Correction Message descriptor and a Control Assign descriptor. In this specific example, it is assumed that the fine synchronization is not achieved yet, even though the return link alignment procedure is successful. The RCST uses the Control SDU slots allocated by the Control Assign descriptor for fine synchronization.

When the fine synchronization is achieved, the NCC sends a TIM-U with Satellite Forward Link and Satellite Return Link descriptors in addition to a Correction Message descriptor. The Satellite Forward Link and Satellite Return Link descriptors correspond to the operational population ID of the RCST. The RCST tunes to the operation TDM FLS and seeks for RA logon slots in the BTP (SCT/FCT2/BCT/TBTP2). It is possible that the NCC may allocate DA logon slots for the RCST to speed up the re-logon process after alignment.

The RCST sends a Logon SDU with `entry_type`==0x1 and the access_status having '1' as the least significant bit.

In response, the NCC sends a TIM-U containing a Logon Response descriptor and a Control Assign descriptor. The Logon Response descriptor assigns all necessary addresses to the RCST since the return link alignment has been achieved. The Control Assign descriptor allocates to the RCST a number of periodic Control SDU slots for maintaining fine synchronization in addition to other return link M&C signalling. The TIM-U may at this stage also contain descriptors necessary to fully configure the RCST.

Note that the NCC may allocate a sequence of DA logon slots in the BTP. If necessary, a much longer superframe duration may be used to allocate longer DA logon slots. Also note that the NCC may terminate the alignment procedure by sending a "failure" in the Pointing Alignment Support descriptor.


RCST
NCC
FWD Alignment
NIT
RMT
TIM-B
Acquisition of operational FLS
SCT, FCT2, BCT, TBTP2
MAC Logon in operational SFS
Logon SDU in RA logon slot
TIM-U
Acquisition of alignment FLS
SCT, FCT2, BCT, TBTP2
Burst-based RTN Alignment
and
Fine synchronisation
Logon SDU with alignment probe in DA logon slot
TIM-U
TBTP2
Logon SDU with alignment probe in DA logon slot
TIM-U
Control SDU
TIM-U
Acquisition of operational FLS
SCT, FCT2, BCT, TBTP2
MAC Logon in operational SFS
Logon SDU in RA slot
TIM-U
Control SDU with CR
CMT or TIM-U::CMD
TBTP2
TRF
RCST
NCC

TIM-B with a Pointing Alignment Support descriptor with alignment_control_type == 0, pointing_alignment_required == 1, burst_based_alignment_supported == 1, and forward_link_snr_threshold. Iterative update of RCST antenna alignment parameters by the installer until forward link SNR is measured to be below forward_link_snr_threshold.

Operational FLS searched in RMT using operational population ID.

entry_type == 0x0, indicating pointing alignment support request. Least significant bit of access_status is 0, indicating that NCC has not confirmed an earlier pointing alignment.

TIM-U with a Logon Response descriptor with RCST_access_status==0x00 and two Pointing Alignment Support descriptors. The first with alignment_control_type == 64 indicating a different alignment population ID. The second with alignment_control_type == 66 indicating burst-based alignment probe pattern.

Alignment FLS searched in RMT using alignment population ID. TBTP2 contains dedicated logon slot allocation for RCST.

Dedicated logon burst with alignment probe.

TIM-U with a Pointing Alignment Support descriptor with alignment_control_type == 96 with alignment_status == 0 (in progress) including CNR, co-polar and x-polar readings at the NCC. TIM-U also contains Correction Message Descriptor.

TBTP2 contains dedicated logon slot allocation for RCST.

Dedicated logon burst with alignment probe.

TIM-U with a Pointing Alignment Support Descriptor alignment_control_type == 96 with alignment_status == 1 (success) including CNR, co-polar and x-polar readings at the NCC. In addition TIM-U contains a Correction Message descriptor and a Control Assign descriptor.

TIM-U contains a Correction Message descriptor indicating achievement of fine synchronisation. In addition, TIM-U contains a Satellite Forward Link and a Satellite Return Link descriptor corresponding to operational population ID for the RCST.

Operational FLS. TBTP2 may contain dedicated logon slot allocation for RCST. Otherwise, RA slot is used for logon.

Dedicated logon burst. entry_type == 0x1, binding user to HW and network. Least significant bit of access_status is 1, indicating that NCC has confirmed pointing alignment.

TIM-U may contain a Logon Response descriptor and Control Assign Descriptor indicating dedicated Control SDU slot allocations that the RCST may use to maintain fine synchronisation in addition to other return link M&C signalling.

Fine synchronisation is achieved and the Control Burst contains Capacity Request.

Dedicated time slot allocations from NCC in TBTP2. User traffic from RCST in TRF slots

Figure 7.3: Example message exchange with IB-based return link alignment, a different population ID for operational and alignment phases

8 Transmission Security Implementations

8.1 Introduction

This clause describes three TRANSmission SECurity (TRANSEC) implementations for the DVB-RCS2 waveform, each covering a different market or profile emphasis.

8.1.1 Scope of TRANSEC Protection for DVB-RCS2

In the context of DVB-RCS2, TRANSEC protection is defined to include:

- Protection of Channel Activity Information
    - To disguise user traffic volumes in both forward and return channels.
- Protection of Control and Management Information
    - To encrypt or otherwise disguise control plane and management plane signalling and signalling tables in both forward and reverse channels
- NCC and RCST Authentication
    - To ensure that only valid RCST can log-on to a valid NCC and conversely that neither an invalid RCST cannot log on to a valid NCC, or a valid RCST cannot log-on to an invalid NCC.
- Anti-Jam and Low Probability of Intercept
    - These attributes are considered to be at a lower priority with respect to the previous points and are omitted for the present.

The countermeasure techniques commonly used for mitigating the above risks consist of link-layer encryption, authentication and traffic activity concealment / obfuscation (and associated key management). Of all these, the link layer encryption is foremost.

8.1.2 TRANSEC Profiles

The implementations to be described here are based on the following DVB-RCS2 TRANSEC profiles:

- Consumer
- Professional
- Governmental

As the naming suggests, each profile has a different market emphasis in mind, and accordingly supports a different balance of TRANSEC features. However, features from each implementation here (and others) can be combined to create Custom profiles. The TRANSEC profiles discussed here have several common features such as the use of AES-256 algorithm for link layer encryption, but differ primarily in cryptographic modes of operation and approaches to key management and authentication.

8.1.3 Generic Implementation Approach

8.1.3.1 Security Architecture

The DVB-RCS2 waveform introduces the notion of TRANSEC hooks. The "hooks" allow the extension of normative DVB-RCS2 waveform with TRANSEC countermeasures. The hooks include:

- What and where to encrypt.
- What to authenticate.
- Where to introduce dummy traffic.
- New signalling and reserved fields in control messages for TRANSEC management.

The "hooks" introduced to the air interface design should accommodate different security implementations with minimal or no impact to complex intellectual property blocks used for modulation/coding and demodulation/decoding and the DVB-RCS2 HLS [i.4].

8.1.3.2 Link Layer Encryption Hooks

TRANSEC encryption should be applied to both Forward Link (FL) and Return Link (RL) of a DVB-RCS2 network, to all data packets (payloads and headers) and to all signalling, with the following exceptions:

- The predefined synchronization sequences, when used (e.g. burst preambles), should be sent in the clear, in order to allow the synchronization of the demodulators.
- The MODCOD field in the Physical Layer (PL) header of the DVB-S2 frames should be transmitted in the clear in order to facilitate frame demodulation / decoding.
- The CRC trailers in DVB-S2 frames should be sent in the clear.

Link layer encryption should be based on approved algorithms, e.g. AES-256, used in approved modes of operations. For governmental applications the encryption algorithms and the operation modes should be government approved.

As a minimum, the Forward and Return Link should each use one Encryption Key for the Interactive Network. It is also acceptable for individual Remote Terminals, or groups of Remote Terminals, to each possess one or more Encryption Keys for each link.

Continuous-carrier return and mesh links should use the methods defined for forward link transmission.

8.1.3.2.1 Forward Link Encryption

Regardless of the precise technique used for Forward Link encryption, each DVB-S2 frame should carry an *initialization Vector* (or *Initial Counter Value*) that can be used for decryption of the frame.

8.1.3.2.2 Return Link Encryption

In the interest of bandwidth efficiency, Return Link and Mesh Link TDMA transmissions do not need to carry explicit initialization vectors. Instead, unique initialization vectors for these transmissions can be computed from a combination of shared secrets and (extended) identifiers of the superframe, frame and time slot already used in DVB-RCS and known to both the RCST and the NCC. The information necessary to support this can be communicated in the Forward Link using the Return_Link_IV sub-type of the TRANSEC_System_Message descriptor. These initialization vectors can be used irrespective of the precise encryption technique employed.

8.1.3.3 Authentication and Key Management Hooks

It is expected that different TRANSEC profiles can have substantially different requirements and preferences for Authentication and Key Management, e.g. some Governmental profiles may only support manual key entry (using approved key fill devices), need minimal new signalling, whereas others (here) support over the air rekeying (and purging), add more complex signalling. Each of the TRANSEC profiles defines signalling elements for MAC layer transport of the associated messages, for example certificate exchanges and key update commands. This does not preclude other methods of transport of such messages, if allowed and/or preferred by specific implementations. In general, it is expected that authentication and mutual trust is established using standardized PKI/X.509 certificate solicitation and exchange.

In the forward link, authentication and key management messages can be transported in system specific variants of the TRANSEC_System_Message descriptor. System specific variants are identified by certain values of the descriptor_type field.

In the return link, authentication and key management messages can be transported using a specific new protocol type. This protocol type can be used with both RLE and GSE encapsulation. The content of PDU's transported with this protocol type is system specific.

8.1.3.4 Traffic Flow Obfuscation Hooks

On the forward link, traffic activity should be concealed / obfuscated by transmitting dummy (or "chaff") packets in broadcast mode (i.e. with no specific terminal address). A special chaff protocol type is defined for this purpose. The dummy packets should be inserted in a frame when not enough actual data are available to fill up the frame. They should include random data which should be encrypted as actual data.

On the return link the traffic activity should be concealed / obfuscated by transmitting dummy bursts. The dummy bursts should be transmitted in allocated slots, when there is no actual (useful) data to transmit. The payload of the dummy bursts should include random data, encrypted as actual data.

8.2 Consumer Profile

8.2.1 Introduction

The aim of the Consumer profile is to provide confidentiality to the higher layer traffic sent over satellite and to protect against satellite link and subscription piracy.

The higher layer protocol TRANSEC provides:

- **Data protection:** The traffic is encrypted using AES-256, with a traffic key shared by a group of RCSTs. This key is distributed over the air.

- **Key protection:** RCST group's keys are protected in the distribution by encryption by use of a passphrase (based on a password shared between each RCST and the key manager).

Encryption of the signalling system is **not** done in the Consumer profile. Keeping the signalling in the clear eases the installation of the RCSTs, and enables the operator to more easily do e.g. the line-up of the RCST without the knowledge of the RCST's password.

The Consumer profile addresses the general user data security concern for both private and shared/public teleports. This requires offering to the users the option of having the layer 3 PDU encrypted as it is in transit in the DVB-RCS network. Different "user groups" on the network (which may be different organizations entirely, each with their own sets of RCSTs), should be allowed their private encryption keys.

The Consumer profile should also support protection for the DVB-RCS network operator, e.g.:

- Prevent use of subscriptions by others than the legitimate subscriber

- Prevent unauthorized use of RCST hardware

- Thwarting productive use of stolen or "hacked" RCSTs

- Detection and thwarting use of cloned RCST MAC addresses

Thwarting denial of service attacks, with respect to network log-on.

8.2.2 Security Architecture

The layered model for the Consumer profile is shown in Figure 8.1. The Encryption hooks are shown in red.

In particular, on the forward link, performing layer 2 encryption of the unicast GSE PDU, and on the return link, performing layer 1 encryption of the exclusive traffic slots.

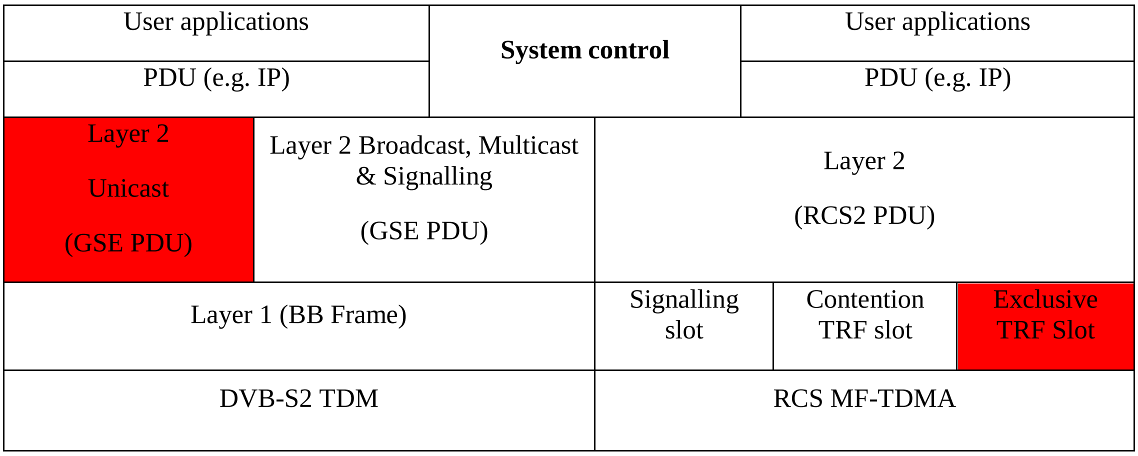


Figure 8.1: Consumer Profile Layered Security Model

8.2.3 Authentication and Key Management

Key management relations between the Feeder, Gateway and RCST are shown in Figure 8.2.

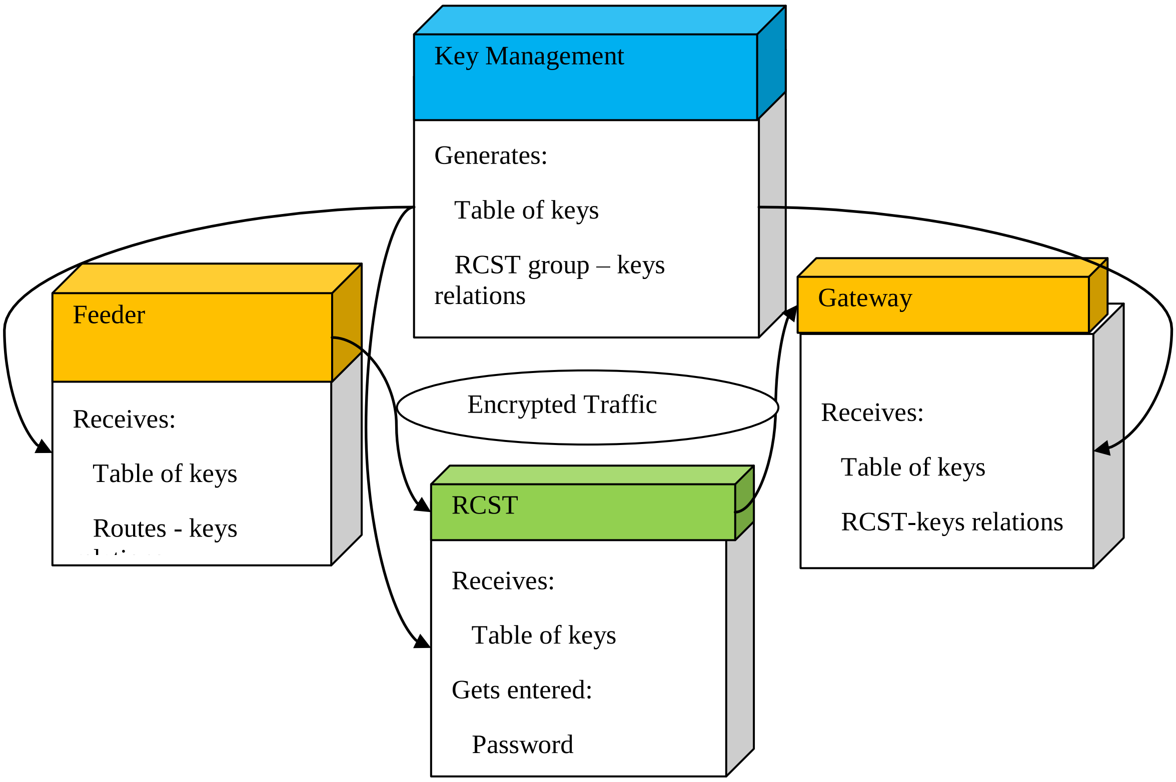


Figure 8.2: Key Management Relationships

8.2.3.1 Security Enabling

- Determine a password for each RCST
- Determine the FL unicast key group of each RCST
- Determine the FL multicast key group(s) of each RCST
- Determine the mesh key group of each mesh RCST
- Generate a traffic key for each key group
- Prepare an encrypted version of the applicable traffic keys for each RCST based on the password
- Enable Feeder and Gateway with the necessary key group keys
- Enter password and switch on TRANSEC operation at each RCST
- Switch on TRANSEC forwarding to all the RCSTs concerned
- Switch on TRANSEC for all the multicast concerned
- Logoff and logon all the RCSTs, so all relevant traffic keys are loaded

8.2.3.2 Security Disabling

Switch off TRANSEC for the Feeder. Switch off TRANSEC for each RCST.

8.2.3.3 Passphrase generation for RCSTs

The passphrase is set by the user. It is used to encrypt the traffic key for the RCST provided at logon.

8.2.3.4 Key renewing

Provision the new key at Feeder, Gateway, and in encrypted versions at NCC, and have the RCSTs logon again.

8.2.3.5 Key encryption

The key encryption is done using AES-256 in CBC mode [i.21] and with the initialization vector is according to clause 8.2.4.5.2.2. The key used for encrypting the traffic key is generated by using the passphrase.

8.2.3.6 RCST key generation

The key used for encrypting traffic keys for a particular RCST is generated from the RCST's passphrase using a predefined algorithm.

The algorithm is assumed agreed between the customer and the equipment vendor.

8.2.3.7 Key ID

The key ID is an index referring to a specific traffic key. A traffic key can be identified by its key ID.

8.2.3.8 Key Storage in the RCST

The RCST acquires the traffic keys at logon and is not expected to store the keys when powered off. It should purge all previous traffic keys when logging on.

8.2.4 Encryption

8.2.4.1 TRANSEC M&C

The TRANSEC M&C interface should only be accessible via a locally connected console or via a non-detachable panel, and not via the IP network. This will prevent illegitimate users from connecting via the IP network.

Set TRANSEC password

This function enters the TRANSEC password for non-volatile storage of data to retrieve the deducted passphrase.

Display TRANSEC status

This function should display the current TRANSEC status. The TRANSEC state should be clearly indicated to the user in a consistent and permanent manner.

Clear TRANSEC password and passphrase

This function clears a user-entered password and the associated passphrase.

8.2.4.2 TRANSEC Procedures

8.2.4.2.1 RCST Logon

The logon procedure is the same as for the non-TRANSEC system, excepted that the RCST will also retrieve its traffic keys as part of the procedure. Upon logon, the traffic keys for the RCST, encrypted by use of the RCST's passphrase, is sent to the RCST via a dedicated descriptor in the unicast TIM. This TIM descriptor is further described in clause 8.2.4.5.1.

8.2.4.2.2 Transmission of encrypted traffic to the Gateway

A TRANSEC enabled RCST encrypts higher layer user packets before transmission.

An Initialization Vector is generated, following the Initialization vector format defined in clause 8.2.4.5.2.1.

The RCST encrypts the payload using the Initialization Vector and the current key, and AES-256 in CTR mode [i.21]. A part of the IV is included in a header extension in the transmitted layer 2 payload, specified in clause 8.2.4.5.3.

The payload is associated with the protocol type of the header extension for higher layer encryption. An explicit indication of this protocol type is included in the payload if the associated default protocol type is not equal to this protocol type.

8.2.4.2.3 Reception of encrypted traffic from the Feeder

A TRANSEC enabled RCST detects the received encrypted higher layer packets and de-crypts before forwarding to the user, using the key explicitly referred to by the Key ID in the extension header, or the default key for the forward link if the Key ID is not indicated. The decryption is equal to the encryption done by the Feeder as described in clause 8.2.4.3.1.2.

After removal of the extension header and decryption the RCST forwards the clear-text packet to the higher layer.

8.2.4.2.4 Decryption of the encrypted traffic key

To decrypt an encrypted key, the RCST uses its own key encryption key that is generated by using the passphrase, and the Initialization Vector associated with the encrypted key. Decryption is equal to the encryption described in clause 8.2.3.5.

8.2.4.3 TRANSEC by the Feeder

8.2.4.3.1 Procedures

8.2.4.3.1.1 Transmission of a unicast packet

If TRANSEC is enabled for the RCST that the unicast packet is aimed for, the Feeder encrypts the higher layer packet before transmission to this RCST.

8.2.4.3.1.2 Encryption of a unicast packet

The Feeder uses an appropriate group traffic key associated with the destination RCST. It generates and uses an Initialization vector following the Initialization vector format defined in clause 8.2.4.5.2.1.

The packet is encrypted using the appropriate Initialization Vector and the unicast traffic key of the RCST, and uses AES-256 in CTR mode [i.21]. A part of the IV is included in a header extension in the transmitted layer 2 payload, specified in clause 8.2.4.5.3. The Feeder will include the Key ID in the header extension when another key than the default is used, and may omit the Key ID if the default key is used.

8.2.4.3.1.3 Transmission of a multicast packet

If TRANSEC is enabled for the multicast group, the Feeder encrypts the higher layer packet before transmission.

8.2.4.3.1.4 Encryption of a multicast packet

The Feeder uses an appropriate group traffic key associated with the multicast group. It generates and uses an Initialization vector following the Initialization vector format defined in clause 8.2.4.5.2.1.

The packet is encrypted using the appropriate Initialization Vector and the applicable key, and uses AES-256 in CTR mode [i.21]. A part of the IV is included in a header extension in the transmitted layer 2 payload, specified in clause 8.2.4.5.3. The Feeder will include the Key ID in the header extension when another key than the default is used, and may omit the Key ID if the default key is used.

8.2.4.4 TRANSEC by the Gateway

8.2.4.4.1 Procedures

8.2.4.4.1.1 Reception of an encrypted packet

Upon reception of an encrypted traffic packet, the Gateway detects that the packet is encrypted, and decrypts it with the appropriate key before forwarding it to the higher layer.

8.2.4.4.1.2 Decryption of a packet

The Gateway selects the applicable traffic key for each encrypted packet received. The traffic key is the one associated with the RCST. This is a key administratively associated to the RCST that sent the packet. This RCST is identified by the burst time plan. The decryption procedure is identical to the encryption procedure used by the RCST as specified in clause 8.2.4.2.2.

8.2.4.5 Format and syntax of TRANSEC elements

8.2.4.5.1 Providing the traffic key in TIM-U

The tag used for this descriptor is recommended to be 0x21.

The descriptor has the following format.

Table 8.1: Key Descriptor used in TIM-U

Syntax	Value	Number of Bytes	Information Mnemonic
Traffic_key_descriptor() {			
Tag	0x21	1	Uimsbf
Length	0x27	1	Uimsbf
Key_use		1	Bslsbf
Random_field		4	Uimsbf
Key_id		2	Uimsbf
Encrypted_key		32	Uimsbf
}			

Description of parameters:

Tag	0x21
Length	0x27
Key_use	control bits defined in Table 8.2
Random_field	32 bit random variable used as part of IV
Key_id	16 bit key identifier
Encrypted_key	256 bit encrypted traffic key

Table 8.2: Administrative Encryption Context Control

Bit	Control Flag	Encoding
0	Forward_link_default	1= yes, 0=no
1	Return_link_use	1= yes, 0=no
2	Mesh_link_use	1= yes, 0=no
3-7	Reserved	

The control flags indicate the contexts where a key may be used. If there are several keys allowed used for the same transmission context, the RCST is free to choose which to use. One key may be used for all the crypto contexts.

TIM-U can contain several keys. If ambiguity is to be avoided, the RCST can be given a single key for each transmission context.

8.2.4.5.2 Initialization Vectors

8.2.4.5.2.1 IV used for traffic encryption

The initialization Vectors used for traffic encryption (both on RL and FL) are generated with the following format: Date of the Day + base of the NCR value. This ensures that the IVs will be changed for every transmitted packet.

Table 8.3: Initialization Vector for encryption of traffic

Syntax	Number of Bytes	Information Mnemonic
TrafficEncryptionInitializationVector() {		
Year	2	Uimsbf
Month	1	Uimsbf
Day	1	Uimsbf
NCR_base	4	Uimsbf
Group_id	1	Uimsbf
Logon_id	2	Uimsbf
Zero-Padding	5	'0'
}		

Description of the parameters:

Year Current year of the operating system encoded in a 16 bit field

Month Current month of the operating system (1-12) encoded in an 8 bit field

Day Current day of the operating system (1-31) encoded in an 8 bit field

NCR_base 32 least significant bits of the local NCR base value sampled just in time before each encryption

Group_id The assigned Group ID of the RCST

Logon_id The assigned Logon ID of the RCST

At the receiver side, the NCR_base value used at encryption is reconstructed by getting the NCR_base LSBs that the transmit side has appended to the encrypted packet, take its own NCR_base value, and then recover the full NCR_base that the transmit side did use by applying the following algorithm:

- Sample the local NCR_base and compare the MSB of the received NCR_base subsection with the locally generated counterpart.

- If the local one is the lowest, then the subsection of the local NCR_base has wrapped relative to the one received with the packet. Subtract 2^(the number of bits of the NCR_base subsection transmitted +1) from the local NCR_base sample, and then replace the LSBs by the received NCR part.

- Else, just replace the LSBs of the local NCR_base sample by the NCR subsection received with the packet.

Similar processing applies as well for Date if the Date value indicates wrapping, and may then also apply for Month and Year. In-band signalling information is used to resolve synchronization issues between the hub and RCST when time-of-day clocks are slightly out of synchronization.

8.2.4.5.2.2 IV used for key encryption

The initialization vector used for key encryption is a 128 bits sequence generated following the structure below.

Table 8.4: Initialization Vector for encryption of a traffic key

Syntax	Number of Bytes	Information Mnemonic
KeyEncryptionInitializationVector(){		
Key_id	2	Uimsbf
Group_id	1	Uimsbf
Logon_id	2	Uimsbf
Random_field	4	Uimsbf
Zero-Padding	7	'0'
}		

Description:

Key_id 16 bit key identifier

Group_id The assigned Group ID of the RCST

Logon_id The assigned Logon ID of the RCST

Random_field As transmitted in the traffic key descriptor

The IV is renewed for each key encrypted, which means that each time the NMS encrypts a new key it also generates and uses a new IV (in effect a new random part and a new key_id for a given RCST).

8.2.4.5.3 Crypto Context Extension Header

A new `Protocol_Type` (2B) value for GSE indicates that there is an `Explicit_Crypto_Context_Extension` header appearing first in the FL payload, followed by an encrypted packet of the indicated protocol type.

Table 8.5: Crypto context extension header applied when using explicit key selection

Syntax	Number of Bytes	Information Mnemonic
Explicit_Crypto_Context_Extension(){		
Key_id	2	Uimsbf
Encryption_Context	3	Uimsbf
Compressed_Protocol_Type	1	Uimsbf
}		

Description of the parameters:

Key_id This 2 byte field indicates the Key ID of the key used to encrypt the packet following the extension header.

Encryption_Context This 3 byte field contains for the FL the 5 LSB bits of Day as MSB and then 19 LSBs of the NCR subsection used in the Initialization Vector as LSBs. This 3 byte field contains for the RL the 24 LSBs of the NCR subsection used in the Initialization Vector.

Compressed_Protocol_Type Compressed protocol type value for the clear-text variant of the encrypted PDU, as specified in EN 301 545-2 [i.1].

The encrypted packet is appended following this extension header.

In the return link and on mesh links, a smaller header extension is always used, as the key ID is determined implicitly. A new Compressed_Protocol_Type (1B) value for RLE indicates the presence of the Implicit_Crypto_Context_Extension. This smaller header extension may also be used in the forward link when encrypting with the key indicated to be the default for the forward link. In this case, a further new Protocol_Type (2B) for GSE indicates the presence.

Table 8.6: Crypto context extension header when using implicit key selection

Syntax	Number of Bytes	Information Mnemonic
Implicit_Crypto_Context_Extension(){		
Encryption_Context	3	Uimsbf
Compressed_Protocol_Type	1	Uimsbf
}		

Encryption_Context This 3 byte field contains for the FL the 5 LSB bits of Day as MSB and then 19 LSBs of the NCR subsection used in the Initialization Vector as LSBs. This 3 byte field contains for the RL the 24 LSBs of the NCR subsection used in the Initialization Vector.

Compressed_Protocol_Type Compressed protocol type value for the clear-text variant of the encrypted PDU, as specified in EN 301 545-2 [i.1].

The encrypted packet is appended following this extension header.

8.2.5 Traffic Obfuscation

The Consumer profile could use DVB-RCS2 Free Capacity Assignment (FCA) capability to ensure that return link carriers and timeslots are maximally filled, in order to prevent traffic analysis based on network activity. Use of FCA ensures that RCSTs will have access to all network capacity at all times to send any available traffic.

8.3 Professional Profile

8.3.1 Introduction

The Professional profile is for markets where comprehensive, strong network security is a primary concern.

It is expected that DVB-RCS networks in these markets will be private networks (with a private NCC/hub system) where the same high grade of security is provided throughout the network.

Requirements for the Professional profile include:

- Obfuscation to prevent traffic pattern or activity analysis on Forward link carriers, which implies the use of:
    - Encryption of all Layer 2, Layer 3 and higher layer information (include all related headers in transparent and mesh overlay systems. In mesh regenerative some headers or specific field may be in clear for OBP routing)
    - Filling of any unoccupied DVB-S2 frames with "chaff"
    - Encryption of all network signalling (excepting the NCR)
    - The option of operator choice regarding the level of traffic obfuscation on return link for mesh systems
- Encryption of payload in each burst (including signalling bursts), and any continuous transmission on these same carriers, protecting all Layer 2, Layer 3 and higher layer information
- Strong authentication:
    - of the Hub/NCC (to the user)
    - of the user (to the Hub/NCC)
    - between users (in mesh systems)

8.3.2 Security Architecture

Figure 8.3 shows the security components of the Professional profile network.

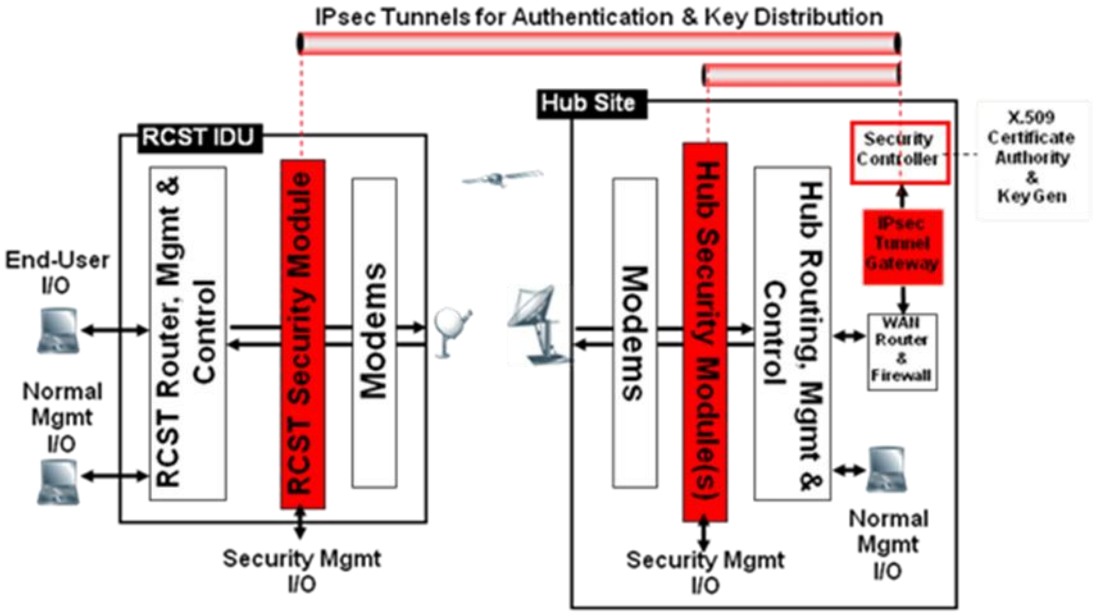


Figure 8.3: Professional Profile Security Architecture

Within the Indoor Unit (IDU) of each RCST there should be security functions of encryption, decryption, authentication and related security management tasks that all are built to a sufficiently qualified trust level. An isolated implementation of these functions in a qualified security module should exist, and here in a module called the **RSM** (for **RCST Security Module**). This module may be built-in at the factory, or provided as a user plug-in option for a more generic HW platform.

At the Hub one or more similar hardware modules should exist, here called **HSM** (for **Hub Security Module**). Typically, one HSM may be required for the Forward Link and another for the Return Link.

For operation on the network, each RSM and each HSM is equipped with its own X.509 Private Key User Certificate, plus the corresponding Root CA Public Key Certificate for the issuing Certificate Authority (CA). Each RSM and each HSM also has IPsec Tunnel Mode capability and mutual authentication capability.

Each RSM and each HSM has local I/O on the module for secure management and operation. This I/O should allow the X.509 Certificates to be conveniently installed, the entry of usernames and passphrases, and other security-related management task that should be done locally.

At the Hub site, there is an **IPsec Tunnel Gateway** which has a corresponding qualified trust level. That can be a standard commercially available piece of equipment. This device should also have an X.509 User Certificate installed with the corresponding Root Certificate from the issuing CA.

The same CA is assumed used to issue the certificates of all these modules. Also, it is likely these modules will have to be compliant with national government standards on information security.

Behind the IPsec Tunnel Gateway resides the **Network Security Controller (NSC)**. This administrative module, based typically on a standard server, controls and supervises the security related status of each user terminal and electronically distributes the traffic keys required for use of the Protected Channel. Remote access to NSC is only possible via IPsec Tunnels from authorized security modules (RSM and HSM), which serves to authenticate the identity of the device at the end of the IPsec tunnel, and also encrypts the communications for privacy and integrity, plus there is protection against replay attacks.

Finally there is the CA itself. It may reside at the Hub site, or elsewhere, and may be accessed via any acceptable method (electronic or otherwise) per policies of the user organization. More details on the functions of each module are provides in Table 8.7.

Table 8.7: Description of Modules for the High Grade Security profile

Security Component	Description
RCST Security Module (RSM)	This is a module with a qualified trust level. It has IPsec tunnel capability for authentication and encryption. It intercepts Layer 2 control signalling. It controls the forwarding of traffic to the internal router with respect to chosen security policy and current security state. It communicates with the Network Security Controller via IPsec tunnels, to decides the current security state of the RCST. It has dedicated local I/O separated from the other parts of the RCST. It locally manages the use of traffic keys for access to the Protected Channel (see next section) which are distributed to it by the Network Security Controller. It synthesizes the traffic key for the Acquisition Channel from external local input from Security Administrator.
Hub Security Module(s) (HSMs)	Similar to RSM, but on the hub side. Typically one per Forward Link carrier and one for a group of carriers supporting the Return Link.
IPsec Tunnel Gateway	This is a module with a qualified trust level. It restricts the communication to/from the Network Security Controller with the RSMs and HSMs to be via mutually authenticated IPsec tunnels, preventing spoofing and replay attacks.
Network Security Controller	This is an administrative unit controlling and supervising security via the RSMs and HSMs. It distributes the necessary crypto material (e.g. the symmetric keys with their applicable cryptoperiods) for the Protected Channel between the RSMs and HSMs. It also implements roll-over control for these traffic keys, and removes lost or compromised entities from the key distribution.
CA and Key Generator	This is a system with a qualified trust level, required for the provisioning of X.509 Certificates. It may also generate all traffic keys used in the network in addition to the conveyor keys (KEK).

8.3.2.1 Encrypted Communications Channels

Symmetrically encrypted communications of user traffic and signalling occurs between the RSM and HSM modules. This is accomplished using **two** duplex communications channels called: the **Acquisition Channel** and the **Protected Channel**. Their purposes, and the reason for using two such channels, are as follows:

- **Acquisition Channel:** For initial acquisition of the Forward Link by RCST, plus terminal log-on, authentication and key distribution, as may generally be required prior to accessing the Protected Channel.

- **Protected Channel:** For protected transport of user traffic and for RCST-NCC communications in mesh systems.

In general, higher protection is provided to the Protected Channel. Breach in the Acquisition Channel encryption should not give access to the Protected Channel. Traffic keys for access to the Protected Channel are provided via the authenticated IPsec tunnel endpoints, and the keys may also be encrypted end-to-end from the Key Generator to the security module.

On the Forward Link, each of these channels is implemented as one input stream in a multi-stream DVB-S2 TDM. On the Return Link both channels have a quantity of TDMA timeslots associated with them (of either variable or fixed capacity). For the Protected Channel, there may also be Mesh Link capacity on TDMA carriers.

The reasons for encrypting the Acquisition Channel are:

- Authenticate users attaching to the network before providing the protected mode traffic keys, which are issued "just-in-time".

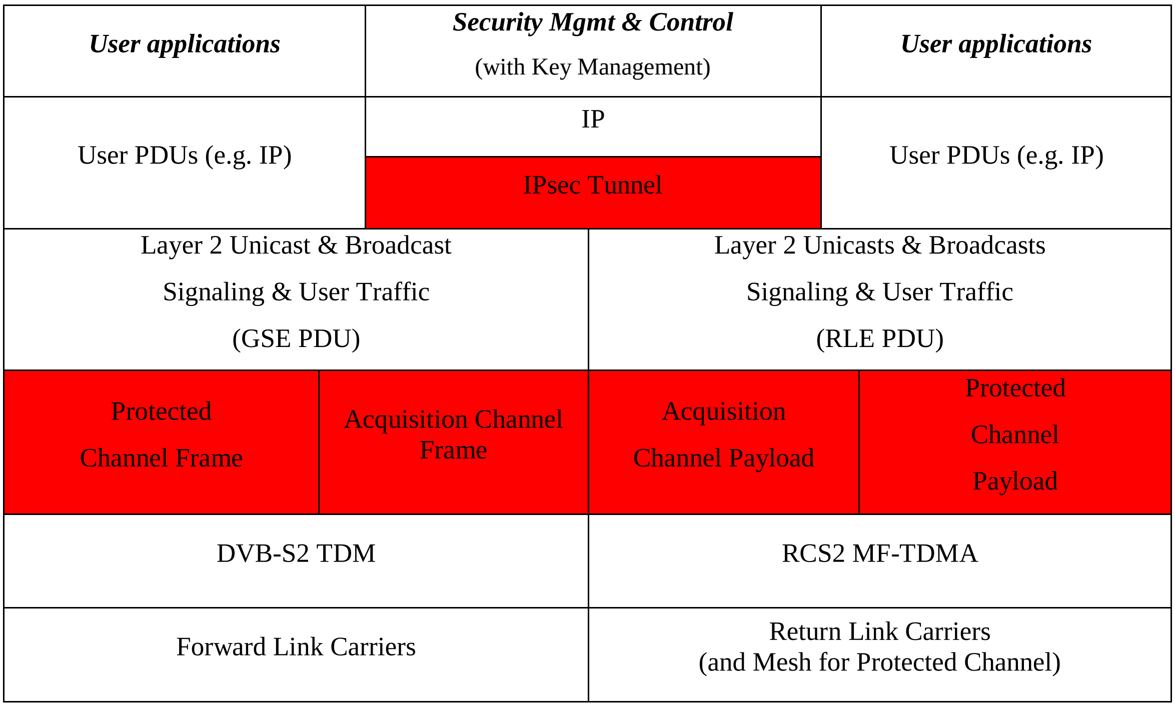
- Conceal information about network identity as may be revealed in SI table content sent over the Acquisition Channel.

- Extend protection against traffic pattern and activity detection to the Acquisition Channel.

The Professional profile requires RCSTs to support encryption on the Acquisition Channel. Some operators may, by policy, choose not to encrypt it, but they would have to ensure that the RCSTs (with their RSMs) selected for their network actually allow for a non-encrypted Acquisition Channel.

8.3.2.2 Layered View of Architecture

Figure **8.4** shows the layered architecture used for the Professional profile. The red blocks indicate where the encryption is applied (i.e. the encryption hooks).



<table>
<tr><td colspan="2"><b><i>User applications</i></b></td><td colspan="2"><b><i>Security Mgmt & Control</i></b>
(with Key Management)</td><td colspan="2"><b><i>User applications</i></b></td></tr>
<tr><td colspan="2" rowspan="2">User PDUs (e.g. IP)</td><td colspan="2">IP</td><td colspan="2" rowspan="2">User PDUs (e.g. IP)</td></tr>
<tr><td colspan="2">IPsec Tunnel</td></tr>
<tr><td colspan="3">Layer 2 Unicast & Broadcast
Signaling & User Traffic
(GSE PDU)</td><td colspan="3">Layer 2 Unicasts & Broadcasts
Signaling & User Traffic
(RLE PDU)</td></tr>
<tr><td>Protected
Channel Frame</td><td colspan="2">Acquisition Channel Frame</td><td colspan="2">Acquisition
Channel Payload</td><td>Protected
Channel
Payload</td></tr>
<tr><td colspan="3">DVB-S2 TDM</td><td colspan="3">RCS2 MF-TDMA</td></tr>
<tr><td colspan="3">Forward Link Carriers</td><td colspan="3">Return Link Carriers
(and Mesh for Protected Channel)</td></tr>
</table>

Figure 8.4: Layer Architecture for the Professional Profile

Obviously, when encryption is applied at a lower layer it also encrypts the upper layers using the lower layer.

8.3.2.3 Network Clock Reference (NCR)

The NCR is sent as a broadcast in the clear. This is done so that the NCR solution can be the same as for the non-secured DVB-RCS implementation, and so that the broadcast NCR can function as the basis of the counter in CTR mode encryption (as described in clause 8.3.4.6). The NCR alone provides no useful information about the network. No other system information is transmitted in the clear.

8.3.3 Authentication and Key Management

8.3.3.1 Network Operation

This clause describes the operational dependencies of a network with the Professional profile, in temporal fashion from initial set-up to routine use of the Protected Channel with key roll-over.

8.3.3.1.1 Installation of X.509 Certificates

Before the RCST can enter the network, the RCST Security Module (RSM) should be loaded with:

- X.509 User Certificate issued by a Certificate Authority (CA)
- The CA's Root Certificate

The User Certificate should be unique for each RCST.

The Hub Security Modules (HSM) also should be loaded with their unique User Certificates and the associated CA Root Certificate. These certificates are loaded the security management port on the RSM and HSM. Also, the public keys of all RSM should be loaded into the NMC/NCC for each terminal, prior to terminal activation on the network, to support logon authentication.

8.3.3.1.2 Security System Time of Day

The security system Time of Day (TOD) should be set by the operator. This is required to align the symmetric encryption. The accuracy should be according to system policies, but a TOD within a couple of minutes should be sufficient to enter the network.

8.3.3.1.3 Network PASSPHRASE

There is a network PASSPHRASE, common to the network as a whole. It is used to restrict access to the Acquisition Channel. It should be known by the RCST operator and should be entered into the RSM of the RCST to gain access to the network.

8.3.3.1.4 USERNAME

Each RCST and each HSM in the operational network is associated with a unique USERNAME (within the organization) for this installation. The USERNAME is associated with a compatible hardware unit by loading the credentials consisting of at least the user certificate and the network PASSPHRASE. The USERNAME identifies the user that has been given these credentials.

8.3.3.1.5 First Power-on of RCST after Installation

When a newly installed RCST is powered-on, it tunes to the assigned DVB-S2 carrier. Then, using standard Base-Band Header information (which remains in the clear), the RCST, by default, selects the stream corresponding to the Acquisition Channel, where it finds the NCR (in the clear) to reach the state of "Ready for Logon". From this state the RCST could begin the log-on process.

However, prior to any attempts by the RCST to log-on (i.e. prior to sending any logon bursts), the operator should have entered:

- The USERNAME of the installation
- The network PASSPHRASE

The network PASSPHRASE, the NCR and the return link structure are used to generate the decryption key and the encryption key for the Acquisition Channel. After decrypting the Acquisition Channel the RCST processes the SI table information. From this information the RCST discovers where to find the carriers for the Return Link and all necessary structure information for this channel. Among this information the RCST learns how to submit a logon burst to the NCC, in a manner compliant with the network policy. This policy may disallow the use of Slotted Aloha access for logon bursts, and instead have exclusive logon timeslots assigned to each RCST.

8.3.3.1.6 Content and Security of Log-on Information

The RCST forms the content of the control burst, which includes current standard elements, such as its 6-byte hardware ID, plus the following additional information required for the Professional profile:

- 4 LSB bytes of the SHA-1 hash of USERNAME (of the installation);
- A Logon Signature for the control burst.

Every logon request should have a Logon Signature. The Logon Signature should be a chosen minimum number of bytes, taken from the least significant bytes of a Digital Signature. The Digital Signature is a standard type according to public key method (e.g. the one from RFC 4880 [i.23]), signed using the private key of the RCST RSM's X.509 Certificate, and is used to sign the following concatenated information:

- The 6-byte Hardware ID of the RCST
- The USERNAME (of the installation)
- A "Slot Position Identifier" in the context of an extended superframe counter

An extended superframe counter is required, e.g. as elaborated in the Governmental profile, clause 8.4.4.2.3.

The NCC/NMC can verify the authenticity of the logon and the user by checking the Logon Signature using the known public key of the RSM. This allows the NCC to ignore logon requests that are unauthorized. By using a time-variant part in the signature the logon process is protected against replay.

8.3.3.1.7 Synthesizing the Key

The Acquisition Channel traffic key is generated at the RCST based on the network PASSPHRASE. The 256 bit key needed is synthesized using the lower 32 bytes of the SHA-1 hash of the PASSPHRASE, combined with a well-known Initialization Vector (IV).

8.3.3.1.8 Set-up of IPsec Tunnels between RSM and the IPsec Gateway

After logon via the Acquisition Channel, the RSM establishes an IPsec tunnel to the IPsec GW and a connection to the NSC via this IPSec GW. The RSM and IPsec GW require mutual X.509 certificate based authentication to set up the tunnel, so the IPsec GW is authenticated to the RSM and vice versa. This is done according to standard procedures specified for [i.21].

The RCST takes the initiative to set up the connection to the NSC whenever this connection is missing. It stays up all the time during the session. It is not disconnected when switching from the AC to the PC.

It should be noted that the aim of setting up the security association at this layer is to allow use of a general purpose certified implementation of an IPsec gateway to isolate a security controller implementation on the hub side, with less strict requirements for qualified trust. The RSM should intercept and block the traffic when operating in the unprotected mode. It should act as the endpoint of the IPsec tunnel.

8.3.3.1.9 Traffic Key Distribution

In transparent systems, when the secured and authenticated Layer 3 connection is established, the NSC loads the RSM with the Tx/Rx key pair needed for protected mode operation. In mesh systems, this permits the start of DCP logon with the NCC procedure, through the protected channel. The key pair and metadata may be transported in one of several different transport modes:

- In the clear (the layer 3 tunnel is secured)
- Asymmetrically encrypted by the CA private key
- Double encrypted, by CA private key and user public key
- Symmetrically encrypted by a KEK

The key pair is issued with a version number.

In mesh systems, the RCST to RCST dynamic link establishment procedure also should provide a Tx/Rx key pair. The same transport modes are available for mesh dynamic links through the RCST to RCST protected channel.

8.3.3.1.10 Set-up of IPsec Tunnels between HSM and the IPsec Gateway

The HSM takes the initiative to set up a secured connection to the security controller, just as the RSM does after having logged on to the network.

8.3.3.1.11 Traffic Key Distribution to the HSM from the Security Controller

The HSM is configured with the PASSPHRASE and synthesizes the traffic key for the Acquisition Channel just as the RSM does.

The HSM receives the traffic keys for the Protected Channel via the IPsec tunnel just as the RSM does.

8.3.3.1.12 Transition to Protected Channel

Once the RCST has been provided with valid keys for the Protected Channel it may attempt to logon to that channel. The logon is constructed as the logon used in the Acquisition Channel. The RSM will keep the security association with the NSC via the IPsec GW during and after the transition.

The RCST should resort to logon via the Acquisition Channel whenever it does not have the current traffic keys for Protected Channel. If it has valid keys, it may attempt to acquire the Protected Channel directly.

8.3.3.1.13 Key Roll-over on Protected Channel

Keys rollover happens in advance of expiration of the current traffic keys. It is seamless for those already attached to the Protected Channel. The keys are distributed via the IPsec Tunnels in unicast mode to each RSM currently on the Protected Channel, and each active HSM.

The NSC loads future versions of the traffic key (or key pair when splitting Tx & Rx) in advance to support key rollover. These keys are provided by the Key Generator (KG). Key rollover for the Forward Link is driven at the discretion of the NSC.

The key version to be used in the return link may track the version used in the forward link, similar to that proposed in the Government Profile in clause 8.4.3.2. Alternatively, more precise rollover may be done by explicit instruction unicasted to each RCST.

The first half of the CRC32 field is used for an encrypted CRC16 taken over the encrypted payload, and the second half is used for a CRC16 taken over the cleartext information, including all cleartext headers, to protect against link errors. This split protects the router functions against link errors as well as the random errors that occur from the use of incorrect decryption keys.

8.3.4 Encryption

8.3.4.1 DVB-S2 Physical Layer

The DVB-S2 Physical Layer (PL) frames and PL headers should be transmitted in-the-clear to support the demodulator.

8.3.4.2 Encryption BBFRAME Payload

Most of the baseband frame (BBFRAME) payload will be encrypted with a symmetric algorithm. There are some exceptions:

- The BBFRAME header
- The whole NCR frame
- The Crypto Block Header (e.g. 1 byte) described in clause 8.3.4.6.1

8.3.4.3 Base-Band (BB) Header

The Base-Band (BB) frame header (BBHEADER) is transmitted in the clear. The content of the BBHEADER should not allow reliable inferences about traffic activity when traffic obfuscation is enforced. For these reasons, and others explained later, the BBHEADER should have settings as follows.

Table 8.8: BBHEADER settings

	BBHEADER Field	**Applicable variants**	**Value**
MATYPE Field	**TS/GS**	Generic Stream	01
	SIS/MIS	Multiple Input Streams	0
	CCM/ACM	CCM or ACM, as appropriate	0 or 1
	ISSYI	(not used)	0
	NPD	(not used)	0
	RO	As appropriate for the Roll-Off Used	
	ISI (2nd byte of MATYPE)	Distinguishes channels ***(see clause 8.3.4.4)***	
	UPL		0
	DFL	Always use max feasible size for the DFL of the frame	
	SYNC	Indicates encrypted BB payload ***(see clause 8.3.4.6) w or w/o NCR***	
	SYNCD		00
	CRC-8	Calculated as normal	

These settings are within the normal DVB-S2 operation contour. No customization of the BBHEADER syntax or semantics is required.

8.3.4.4 Acquisition and Protected Channels on the Forward Link

The Acquisition Channel and the Protected Channel on the Forward Link are created using the following fields in the BBHEADER.

- ISI byte (Input Stream Identifier, as available with Multiple Input Streams)
- SYNC byte (as is available with Continuous Generic Streams)

The ISI byte is used here to indicate the Acquisition Channel vs. the Protected Channel. The Professional profile could specify a default value for the ISI for the Acquisition Channel to promote interoperability. The ISI byte value for the Protected Channel can be specified via signalling from the NCC, via the Acquisition Channel.

The SYNC byte is used to identify whether DVB-RCS encryption is used or not on the Acquisition Channel. The SYNC byte values for indicating this should be taken from the private range (0xB9 to 0xFF) currently allowed for the SYNC byte, and effort should be made to append to the standardized values.

Two values are needed to cover two cases:

- BBFRAME payload holds an encrypted block (header and payload)
- The NCR (encapsulated in GSE and in-the-clear) immediately follows the BBHeader, and is followed by an encrypted block (header and payload)

The recommended values, to facilitate appending these to standard values are: 0xB9 and 0xBA respectively. (For both formats the encrypted payload of the BBFrame should fill-out the rest of the frame to its maximum feasible DFL value (given the PL layer parameters for that PL frame), leaving room for only a CRC-32 field at the very end of the BBFRAME. This field is split in two, where the first half is used for an encrypted CRC-16 covering the encrypted payload, whereas the last part is used for a cleartext CRC-16 covering the whole BBFRAME, including the header.

Where encrypted, the encryption applies to the content (i.e. the payload) of the BBFRAME.

8.3.4.5 Acquisition and Protected Channels on the Return Link

Implementation of these channels on the Return Link may be accomplished means of resources for the Acquisition Channel and resources for the Protected Channel. An RCST on the Acquisition Channel Forward Link will receive only the signalling applicable to the Return Link for the Acquisition Channel.

An RCST on the Protected Channel Forward Link will receive the signalling from this channel and will stop receiving anything else than the NCR from the Acquisition Channel.

To implement this, some SI that might in unprotected systems be sent as one table, will have to be sent as two tables, one variant in the Acquisition Channel and another variant in the Protected Channel.

The resources may be organized to obfuscate the traffic patterns on the carriers used for the Acquisition Channel and the carriers used for the Protected Cannel, by overlapping these channels on shared carriers.

The encrypted payload of the frame should fill-out the rest of the frame to its maximum feasible value, leaving room for only the CRC-32 field at the very end of the RLE packet. This field is split in two, where the first half is used for an encrypted CRC16 covering the encrypted payload, whereas the last part is used for a cleartext CRC-16 covering the wholeRLE packet, including the header.

8.3.4.6 AES-CTR-256 for Link Encryption

The Professional profiles use AES-256 in Counter (CTR) mode (AES-256-CTR). This allows for variable length blocks of data to be encrypted without compromising security. That results in greater overall network efficiency.

Different variations of this method can be applied to the Protected Channels on the Forward (and Regenerative Mesh), Return (and Transparent Mesh) Link carriers, and the Acquisition Channel on Forward and Return Link. In particular, the application to regenerative mesh systems imposes special constraints (as described in clause 8.3.4.6.2).

8.3.4.6.1 Transparent Operation

For the Forward Link, the cleartext NCR, with an extension method applied, is proposed used as the counter and may be combined with the network PASSPHRASE and other bit string in the application of CTR mode. The native broadcasted DVB-RCS NCR wraps in less than 46 hours and is not sufficient alone. This design utilizes a counter that is already well established in DVB-RCS and avoids including other explicit counters that should be maintained and trusted for the CTR counter purpose.

The NCR extension method is proposed to be a synthesis by a well-known algorithm from the knowledge of the security system Time of Day (TOD), and then aligned with the standard broadcasted NCR at a suitable resolution. Entering the security system TOD with a precision of a couple of minutes should be operationally feasible and should also be sufficient to acquire the link. Better accuracy in entered TOD than that should probably not be required.

One selected byte (the Crypto Block Header) of the NCR is attached to the encrypted data block as header as an explicitly signalled part of the counter to help resolve NCR ambiguity due to uncertainty in NCR reconstruction. The encryption unit chooses one value of NCR from its own NCR counter, and the decryption unit chooses a value of the NCR from its own regenerated NCR as assumed used by the encryption unit (referring to the satellite position). These NCR values are likely to be close but also likely to be slightly different when operating with the required resolution of the NCR section to be used in the counter value.

For the Return Link, the AES-256 counter should be based on an extended superframe counter, combined and other SI table numbers (e.g. timeslot_number, frame_number, superframe ID). The sizes of the bit fields used to form this counter should anticipated future expansion of the frame_number and timeslot_counter to at least full bytes (e.g. 1 and 2 bytes, respectively).

Different keys may be used for the Forward Link vs. Return Link directions **at security policy discretion**. Market demand for this exists today, as it enhances security and allows some added flexibilities, at the expense of distributing more keys.

8.3.4.6.2 Regenerative Mesh Operation

When considering the Regenerative Mesh return link, the use of the implicit cryptographic synchronization techniques like the extended superframe counter is problematic, because when data is demodulated on-board the satellite, the extended superframe counter meaning is lost during multiplexing and re-modulating the data on the processed DVB-S2 downlink. An alternative method for the regenerative return link is to use explicit cryptographic synchronization vector (embedded in the RLE packet), such as a variant of NCR extension (used on the transparent forward link). In this regenerative variant, a shared secret initialization Vector (IV) is used to extend the NCR. This IV can be provisioned in all RCST. The operator may decide to have a common IV for all RCSTs in the network, or per SVN, or even one IV per each RCST pair communicating.

Another possibility for systems that support DCP is to dynamically negotiate the IV using specific TRANSEC protocol DCP messages. This negotiation is implemented with NCC intervention to maintain the secret and an authenticated and trusted IV. The NCC DCP Link Service Establishment_Responses may contain an additional IE with a random IV that would be used for each RCST connectivity link. DCP messages should be encrypted, and therefore the secret is assured.

8.3.4.7 Encryption Used at higher layers

For the IPsec tunnel between SRM and SGW, the following encryption methods are required:

- a digital signature method, e.g. from [i.22]
- a symmetric key encryption method, e.g. from [i.22]

For encryption of the key pair the following methods are required supported:

- an asymmetric encryption method, e.g. from [i.21]

For key distribution from CA/KG, an asymmetric encryption method, e.g. from [i.21].

For key distribution from Key Generator device, a symmetric encryption method e.g. from [i.21].

8.3.5 Traffic Obfuscation

On the forward link, traffic obfuscation is achieved by filling of any unoccupied DVB-S2 frames with "chaff" packets.

Operator discretion should be supported regarding the desired level traffic obfuscation on Return/Mesh carriers. To accomplish full obfuscation it would be necessary to use Constant Rate Assignments (CRA) exclusively, and/or use FCA systematically to complement Bandwidth-on-Demand, and to rule out contention-based SYNC and logon bursts for signalling. This will usually cause a large reduction in usage efficiency, which could have an impact of the cost of operations. This mode also limits flexibility, as terminals cannot easily get significantly more bandwidth in times of sudden need, which could impair accomplishing the mission at a critical time.

It is therefore reasonable to support conditional obfuscation within the context of the Professional profile, to be implemented selectively by network operator.

An enhanced NCC, using robust assignment algorithms and input on current obfuscation policy by terminal, can enforce the necessary rules for traffic activity obfuscation on TDMA carriers using standard DVB-RCS signalling. Thus, there is no need here for a separate security profile to allow selective relaxation on traffic activity obfuscation, and no issues affecting inter-operability.

8.3.6 Regenerative Mesh Extension

Mesh Interactive Satellite Systems use the Professional profile as the standard TRANSEC implementation. This section describes explicitly some extensions needed to implement TRANSEC in conjunction with dynamic connectivity and the DCP protocol. The mesh extension permits also to establish Protected Channels without HSM interaction.

8.3.6.1 Encrypted Communications Channels, layered architecture and NCR

Regenerative Mesh Systems follow the same architecture and procedures as stated in clauses 8.3.2.1, 8.3.2.2, and 8.3.2.3 with the noted exception for mesh.

Additionally professional mesh RCST can establish protected channels with other RCSTs.

Figure 8.5 illustrates mesh extension possible channels.

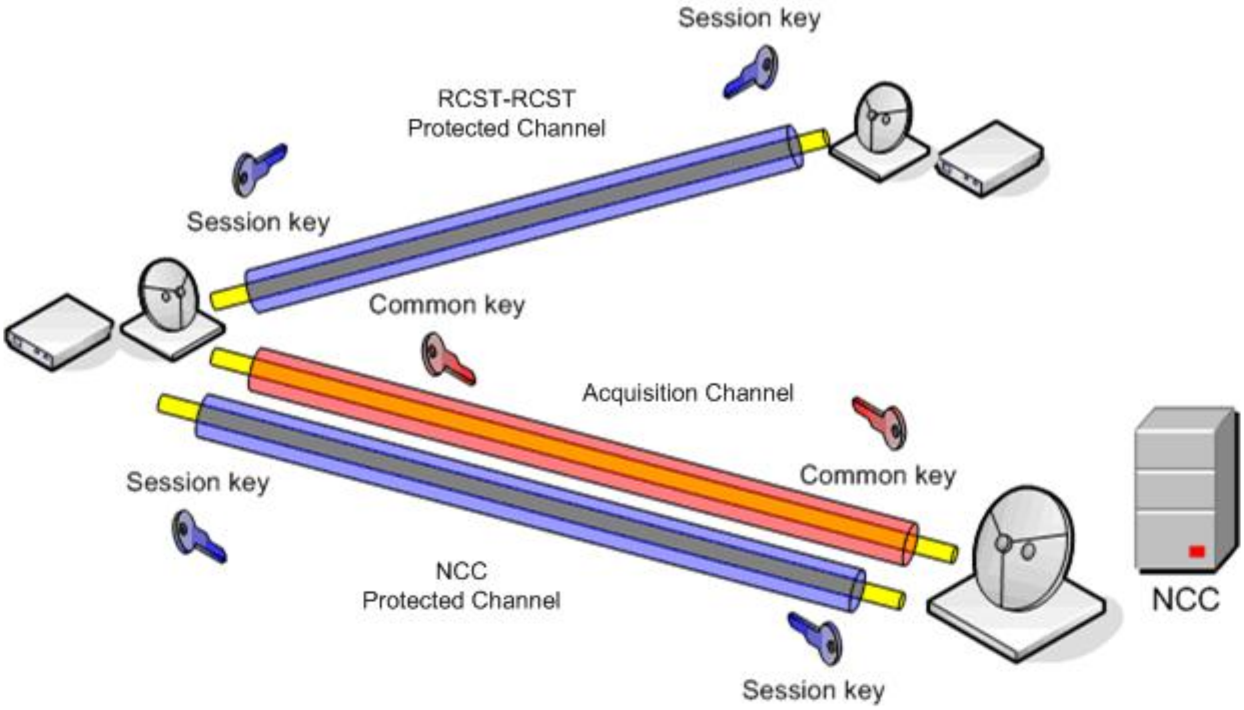


Figure 8.5: Mesh extension in the professional profile

8.3.6.2 Authentication and Key Management

Mesh Systems follow the same procedures as stated in clause 8.3.3 with the noted exception for mesh. These procedures are used to achieve the "System Logon" The DCP Logon and RCST to RCST or RCST to NCC/GW dynamic link establishments are specified in following subsections.

8.3.6.3 DCP Logon

DCP Logon (see the DVB RCS2 HLS [i.4], annex E) should be performed over the protected channel with the NCC. The NCC maintains one protected channel (and one session key) per logged-in RCST. This channel is used for synchronization maintenance and for DCP connection establishment. When the RCST communicates with other RCST a new protected channel is established and the RCSTs communicate using a new session key.

8.3.6.4 RCST to RCST Dynamic Link establishment

This clause describes the connection establishment and the RCST protected channel acquisition with other RCST as per the following procedure using DCP (the DVB RCS2 HLS [i.4], annex E):

1) The RCST starts a communication sending a (bi-directional) Link Service Establishment Request by sending a RCST DCP message to the NCC, using the protected channel. This channel has been obtained using an authentication procedure, so this exchange is considered authenticated and secure.

2) The NCC receives the message and decodes it using the protected channel key.

3) The NCC performs the following functions:

    a) The NCC checks the connection parameters and approves the connection.

    b) The NCC creates the session key. This is an AES 256 key formed from random bytes.

    c) The NCC extracts the destination and source RCSTs X.509 certificates from the NMC database.

    d) The NCC creates two versions of the session key coded with:

        - the destination RSA public key, and

        - the source RSA public key.

4) The NCC sends the Link Service Establishment Request by sending the NCC DCP message to the called RCST, containing, the Connection Control_Descriptor containing an information element (IE) with the session key encrypted with the called RCST public key. This entire message is encrypted with the peer RCST protected key, shared with the NCC so it is also authenticated.

5) The Called RCST decrypts the message with the protected channel key, obtains the connection information and approves the connection. Finally obtains the session key decrypting the IE with its private key.

6) The Called RCST sends the Link Service Establishment Response by sending the RCST DCP message to the NCC using the protected channel.

7) The NCC sends the Link Service Establishment Response by sending the NCC DCP message to the Calling RCST, with an information element (IE), with the session key encrypted with the calling RCST public key. This entire message is encrypted with the peer RCST protected key, shared with the NCC so it is also authenticated.

8) The Calling RCST decrypts the message with the protected channel key, obtains the connection information and approves the connection. Finally, it obtains the session key by decrypting the information element (IE) with its private key.

9) All messages between the RCSTs are exchanged using the session key. The session has been authenticated (as both RCSTs obtained the session key, using its private key, and the NCC is an authenticated entity they trust) and is secure as AES-256 is used.

This case is the most generic one. Other dynamic link establishment types (e.g. uni-directional, NCC initiated) should follow the same procedure of establishing protected channels.

Note that traffic dynamic links are established using an AES session key per traffic connection.

8.4 Government Profile

8.4.1 Introduction

This clause presents a transmission security (TRANSEC) feature implementation for DVB-RCS2. The scheme is based on an existing, approved and tested implementation which is deployed on a point-to-multipoint VSAT system that has many characteristics in common with DVB-RCS(2), including a shared forward link based on DVB-S2 and an MF-TDMA return link. Modifications have been made to the extent required for the TRANSEC scheme to fit in the DVB-RCS2 framework.

Equipment incorporating the scheme on which this proposal is based has been certified in accordance with FIPS 140-2 level 2 [i.34].

The security features offered by this scheme include authentication, link layer encryption and traffic obfuscation. Authentication is based on Public Key Infrastructure (PKI) and X.509 certificates [i.24]. This is considered sufficient for RCST Authentication and Encryption Key Exchange, as well as for protection against cloning, hub faking and replay attacks. The link layer encryption employs AES-256 in approved modes of operation; the security architecture is arranged such that no element of signalling or traffic ever traverses the satellite link unencrypted. The traffic obfuscation feature ensures that all frames in the forward link are filled with encrypted data, so that actual traffic activity cannot be determined by observing the signal. It also ensures that all traffic time slots in the return link are filled with bursts. The occupancy of logon and control slots masks the real logon and synchronization maintenance activity in the network.

8.4.2 Security Architecture

This clause describes the functional architecture of the security system. Other than the separation into Network Control Centre (NCC) and RCST, this architecture does not make any assumptions about the physical implementation of the functional elements. The nature of this clause is informative; specific requirements are introduced in later clauses that deal with the individual security features in more detail.

The architecture is based on the use of two bi-directional, logical channels. These are known as the Acquisition Ciphertext Channel (ACC) and Dynamic Ciphertext Channel (DCC), respectively. These channels are encrypted using separate keys. The DCC is preferred and is used for all unicast and multicast traffic as well as for the majority of signalling exchanged with RCST's that are fully synchronized and authenticated in the network. The Acquisition Ciphertext Channel is used for initial logon, authentication and to exchange keys for the Dynamic Ciphertext Channel. It is also used for any information that needs to be sent to logged-off terminals. The rationale for this arrangement is that, in the event the ACC key becomes compromised; only information about the acquisition process is exposed. The network can continue to operate while a new ACC key is established. Because the DCC key is protected by the RSA public/private keys, possession of the ACC key does not allow an attacker to recover the DCC key. At the same time, this arrangement allows all signalling information to be encrypted, even when exchanged with RCST's that are not logged on.

The air interface used to transport the two channels is described in more detail in clause 8.4.4. For the purpose of this architectural description, it suffices to note that each frame of the DVB-S2 forward link can contain data for either or both logical channels, while each burst in the return link contains data for only one logical channel.

The architecture is described in more detail in the following sub-clauses. Internal interfaces between the functional units are not defined in detail; these interfaces are implementation-specific and can be assumed to carry all necessary side information, such as addressing, priorities, MODCOD selection and choice of logical channel.

8.4.2.1 NCC Security Architecture

The NCC security architecture is shown in Figure 8.6. The architecture works for CCM, VCM and ACM. The functions of the elements are as follows:

1) The external network as shown here encompasses both red and black networks; the interface is thus shown inside a possible convergence router.

2) The firewall may or may not be present in this location, depending on the red-black network configuration. However, this does not affect the TRANSEC architecture significantly.

3) The edge router is the main, bi-directional interface point of the satellite network. As indicated, actual user traffic for the forward link will normally be carried in the DCC.

4) The encapsulation and mode adaptation unit operates in a manner similar to that used in non-TRANSEC operation, whether in CCM, VCM or ACM mode. Its main purpose is to create the payloads of BBFRAME's from the incoming IP traffic and signalling. The unit does however have a number of additional features, in particular:

   - The ability to create place-holders for a number of fields associated with the encryption.
   - The ability to create place-holders for dummy or "chaff" GSE packets to fill any portion of a BBFRAME that cannot be filled with real traffic.
   - The ability to create complete frames filled with chaff packets, so that the transmission of DVB-S2 dummy frames is avoided.

   The air interface is described in more detail in clause 8.4.4. There are logically separate outputs from this unit for the ACC and DCC portions of the frame payload.

5) The ACC "stamping" module performs NCR re-stamping in accordance with the standard, based on SOF time stamps as indicated. It also inserts the initialization vector (IV) value for the encryption in the place-holder provided and fills any chaff packets with pseudo-random data.

6) The stamping module for the DCC is functionally identical to that for the ACC, except that it does not perform IV stamping or NCR re-stamping.

7) Encryption of the ACC is performed independently, using the appropriate key. As indicated, the IV for subsequent encryption operations depends on the output of previous operations.

8) Encryption of the DCC is performed independently, using the appropriate key. As indicated, the IV for subsequent encryption operations depends on the output of previous operations.

9) Construction of the BBFRAME encompasses insertion of the BBHEADER, concatenation of the elements of the frame resulting from the ACC and DCC encryption, and addition of the BBFRAME Cyclic redundancy Check (CRC) in accordance with the standard.

10) The DVB-S2 modulation includes all functions defined in the standard [i.2] from the mode adaptation interface onwards. The resulting transmission is fully DVB-S2 compliant.

11) The DVB-RCS demodulator recovers encrypted return link burst payloads from the IF interface. A practical system usually has several return link carriers.

12) The decryption device recovers the plaintext from each burst payload. Since each burst carries only one logical channel, there is no need for parallel decryption. The channel used is indicated in encrypted form within the burst itself, as described in clause 8.4.4.2.

13) The burst processing function separates traffic, in-band signalling and payloads from logon and control bursts. It directs them to the signalling and traffic processors as appropriate. This function is identical to that found in non-TRANSEC systems.

14) The signalling processing is similar to that found in non-TRANSEC systems; however, it has a number of additional functions necessitated by TRANSEC. The functions of this element include:

   - Generation of "static" signalling for the forward link; this will be carried in the ACC.
   - Processing of signal level, timing and frequency offset measurements passed on from the demodulator, to generate appropriate RCST control messages. These will be passed back to the RCST in either the ACC or DCC, as described later.
   - Handling of initial terminal synchronization and MAC layer logon, as per the standard. This is usually carried out in conjunction with the Network Management System. Following completion of this process (i.e. when the RCST is in the "TDMA Sync" state), the Security Management System is alerted to initiate the Authentication Process.
   - Processing of in-band and out-of-band capacity requests to generate the Terminal Burst Time Plan (TBTP2) for traffic. The TBTP2 differs from that in a non-TRANSEC system in that all traffic slots are always assigned, even in a lightly loaded system. This is part of the traffic obfuscation that hides the actual traffic activity in the overall network and per-RCST. The TBTP2 is separated into two parts; that for the ACC is sent in the ACC; that for the DCC is sent in the DCC.

- Generation of slot assignments for logon and control bursts in a manner that obfuscates the actual activity pattern, as described later.

15) The traffic processing operates almost identically to non-TRANSEC systems. Its main purpose is to re-assemble layer-3 (IP) packets from the fragments received in burst payloads. Normal user traffic is forwarded to the edge router. However, any TRANSEC-related messages (e.g. elements of the authentication exchanges) are forwarded to the Security Management System; they never appear on external interfaces.

16) The Security management System is unique to a TRANSEC-enabled network. The main functions of this unit are:

    - Generation of public and private encryption keys and security certificates or, alternatively, interfacing to external key foundries and/or certificate authorities to obtain these over a secure link. The choice of method is system-specific. The external interface is required for multi-beam systems that support mobile terminal roaming, in order to facilitate handover of RCST's between beams.

    - Distribution of keys within the system (within the NCC and over-the-air to RCST's) and control of key roll-over. Internal distribution is not shown in the diagram, to reduce clutter. In order to support handover of mobile RCST's, an external interface for transmitting or receiving key roll-over commands is required.

    - Management of RCST authentication at logon, communicating directly with the RCST.

    - Informing the network management system of the outcome of the authentication.

    - Operator interface for configuration of all security-related parameters, zeroising compromised RCST's etc.

17) The Network Management System operates in a manner very similar to that in a non-TRANSEC network, including management of QoS etc., which are not security-related. The NMS will however not enable an RCST for regular operation in the Signalling Processing unit until it has passed the authentication process. Should the authentication process fail, the RCST will be logged off immediately.

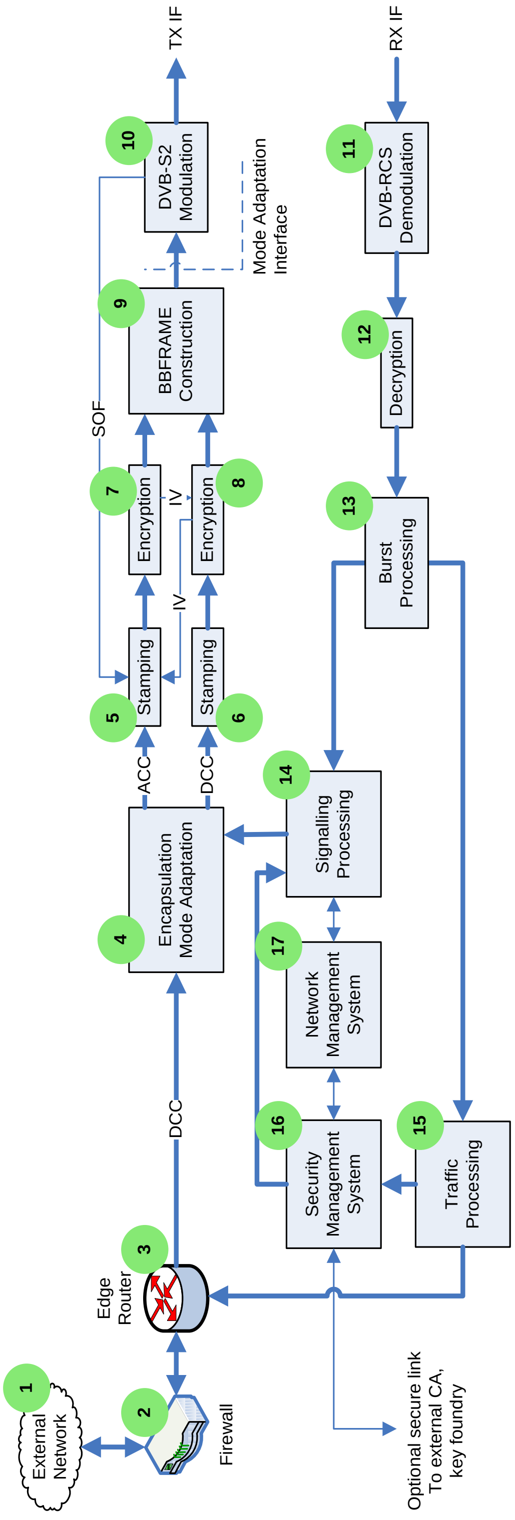


Figure 8.6: NCC security architecture

8.4.2.2 RCST Security Architecture

The RCST security architecture is shown in Figure 8.7. The architecture applies to CCM, VCM and ACM in the forward link and to static as well as adaptive return links. Since the BBHEADER is not encrypted, the scheme is fully DVB-S2 compliant. The functions of the elements are as follows:

1) The DVB-S2 demodulator operates in the same manner as in a non-TRANSEC network. It delivers all the BBFRAME's that it is able to demodulate correctly, based on the BBFRAME CRC check.

2) The stream filter can remove streams in the forward link carrier that are known not to be relevant, based e.g. on the ISI value in the BBHEADER (the BBHEADER is not encrypted). This part of its function is identical to what it does in non-TRANSEC networks. Furthermore, based on information obtained from a partial decryption of the frame using the ACC key, the stream filter separates the frame payload into parts that need to be completely decrypted using the ACC and DCC keys, respectively. It also extracts the IV to be used for decryption.

3) The ACC decrypter operates twice per frame; first to recover plaintext information about the amounts of AC and DC data contained, and subsequently to decrypt the ACC data proper. The DCC decrypter recovers plaintext data for the DCC channel.

4) The output of the decrypters are GSE PDU fragments and/or complete packets. These are provided to the de-encapsulators. The de-encapsulation process is essentially identical to that in non-TRANSEC networks. It re-assembles IP packets from the GSE PDU's and extracts signalling information. It operates separately on PDU's from the ACC and DCC.

5) The signalling processing is identical to that for non-TRANSEC networks, except that it needs to handle TRANSEC-specific signalling elements. These are carried in the normative "hooks".

6) The IP processing is largely identical to that in non-TRANSEC networks. Any IP packets destined for the security management functions (this is not used in the current implementation) are routed internally and never appear on the external interface. As in non-TRANSEC networks, incoming data are sent to the queues/buffers for transmission in the return link.

7) The local (site) network encompasses all local connectivity; specifically, this interface is considered to be inside any separation between black and red networks.

8) The security management function is specific to TRANSEC networks; essentially it is the counterpart of the Security Management System in the NCC. It handles the following:

   - Protocols for all exchanges of certificates and key updates, including generation of security-related messages to be carried in the return link

   - Zeroising, status reporting (not part of current implementation)

   - Encryption and decryption of messages encrypted using the public-key scheme

   - Distribution of keys to the encryption and decryption units in the RCST and key roll-over (not shown in the diagram to reduce clutter)

9) The traffic processing subsystem has the functions also found in non-TRANSEC systems, including separation into traffic/request classes, fragmentation and encapsulation. The encapsulation process is altered slightly, to account for burst payload sizes being modified to allow room for encryption-related parameters. The buffering function contains the necessary classification to separate packets to be sent using the ACC and DCC respectively.

   It should be noted that it will in general not be necessary to replicate the complete set of QoS classifications in the ACC: This channel is never used for actual user traffic, and its use is limited as soon as the authentication is complete and the keys have been established for the DCC.

10) Capacity requests are generated as in non-TRANSEC systems; the only difference being that the request generation needs to consider also traffic queued in the ACC.

11) The burst formatting function generates bursts in all time slots assigned in the TBTP2. If there is no actual content to transmit, a "chaff" payload is generated. This function also includes insertion of encryption-related parameters and in-band capacity requests. The channel (ACC or DCC) to use is determined by the source of the burst payload at the head of the queue in the buffering sub-system.

12) The complete burst payload is encrypted using the appropriate key(s), as described in detail in clause 8.4.4.2.

13) The modulation function is identical to that in non-TRANSEC networks.

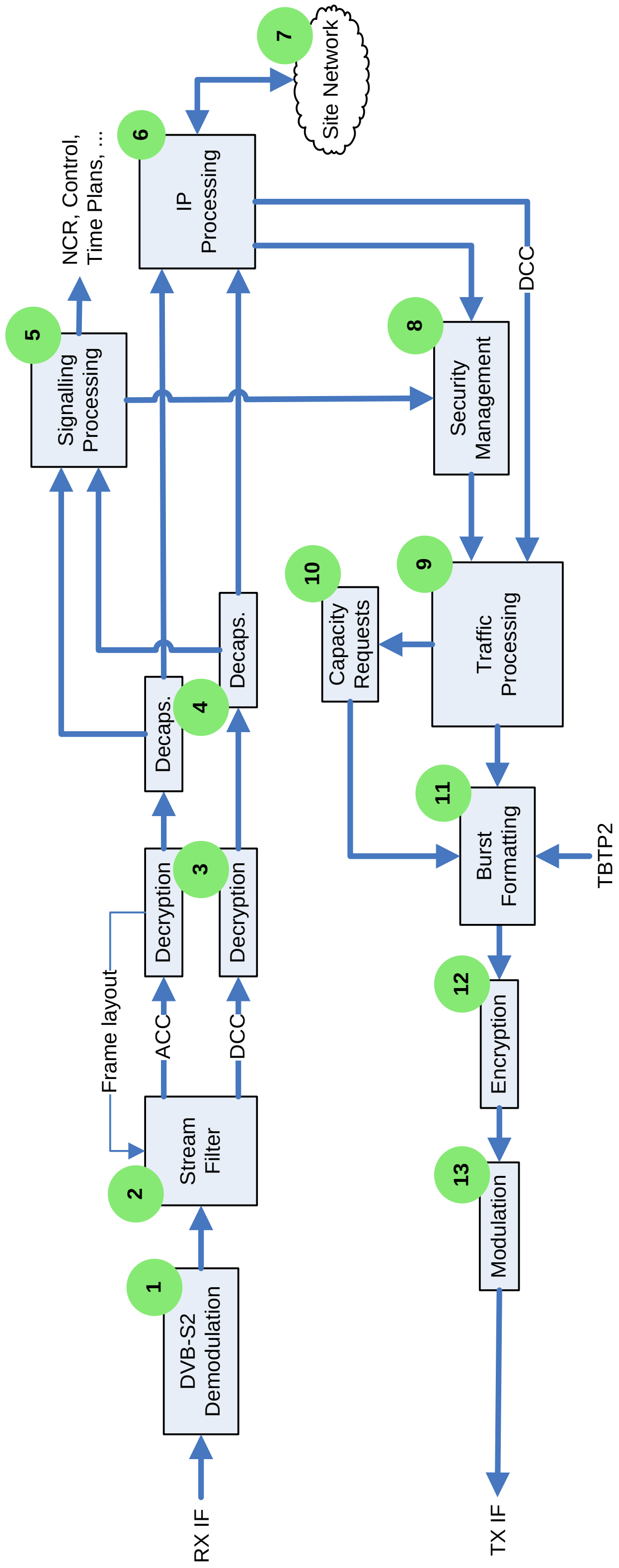


Figure 8.7: RCST security architecture

8.4.3 Authentication and Key Management

8.4.3.1 Authentication of RCST's and Hubs

The following steps constitute the complete logon and authentication process for RCST's. While most of the authentication takes place following the physical/MAC layer log-on and synchronization, certain actions are taken in advance in order to protect the corresponding information:

1) The NCC generates 2 TBTP2's per superframe. One of these is the normal time plan used to indicate to RCST's the slots in which they may transmit. This TBTP2 plan is always encrypted using the active DCC key. The second TBTP2 is encrypted using the network acquisition (ACC) key. It contains assignments of logon slots (if used) and also of other selected slots used for authentication purposes. The union of the two TBTP2's covers all slots within a superframe.

2) The TBTP2's are forwarded and broadcast in the normal manner. RCST's that are not yet acquired receive the time plan via the ACC and identify a suitable logon slot to initiate the physical-layer log-on. This can be a random-access or an assigned slot as defined in the normative document.

3) The RCST acquires the return link in the normal fashion, using a sequence of logon and control slots. All assignments for these slots are carried in the TBTP2 and/or TIM contained in the ACC.

4) Once the RCST has reached the "TDMA Sync" state, it should follow the steps of the key distribution protocol described in clause 8.4.3.2 before it is trusted by the network, and for it to trust the network. All communications associated with this are carried in the ACC. Hence, RCST's in this state will request capacity normally and will be granted TDMA slots required for the key distribution exchanges. During this step, the hub and RCST exchange key negotiation messages in the ACC. Three message types exist: solicitations, certificate presentations and key updates. Solicitation messages are used to synchronize, request, inform, and acknowledge the peer. Certificate presentations contain X.509 certificates. Key updates contain AES key information, signed and encrypted with RSA-2048 [i.25]. The RSA encryption is done using the RCST's public key and the signature is created using the NCC's private key. The protocol messages are defined in clause 8.4.3.4.

5) Once authentication is complete, the key update message should also complete in the ACC. The actual symmetric keys are encrypted using the RCST's public key information obtained in the exchanged certificates.

6) Once the symmetric key is exchanged, the RCST enters the network as a trusted entity and begins normal operation, using the DCC for all traffic and signalling.

The admission procedure outlined above ensures that neither control information nor actual traffic is ever allowed to traverse the air interface unencrypted.

8.4.3.2 Key Management

The key management protocol defined in this clause meets FIPS 140-2 requirements for PKI-based key management protocols that use X.509 [i.24] certificate authentication. When the present TRANSEC implementation is employed, this protocol is mandated for authentication of RCST's, and is recommended also for all other key management operations within the system - for example, the transfer of keying material between the Security Management System and the actual encryption devices within the gateway, or exchanges with external certificate authorities and key foundries.

There are three elements of this protocol: Key distribution, Key rollover and Host Keying.

8.4.3.2.1. Key Distribution

The key distribution protocol is illustrated in Figure 8.8. This protocol assumes that, upon receipt of a certificate from a peer, the host is able to validate and establish a chain of trust based on the contents of the certificate. This allows the NCC to authenticate the RCST, and also allows the RCST to authenticate the NCC, thus preventing it from being "hijacked" by a hostile NCC. Certificate formats and methodologies to verify the peer's certificate are in accordance with X.509 [i.24]. When used as part of the authentication of an RCST, the first certificate solicitation will normally be initiated by the NCC (specifically, the Security Management System).

When the present TRANSEC implementation is employed, the key update is a mandatory part of the protocol when used as part of the authentication of an RCST. Key updates may also be initiated by either peer at any time. Details of the protocol are defined in clause 8.4.3.4.

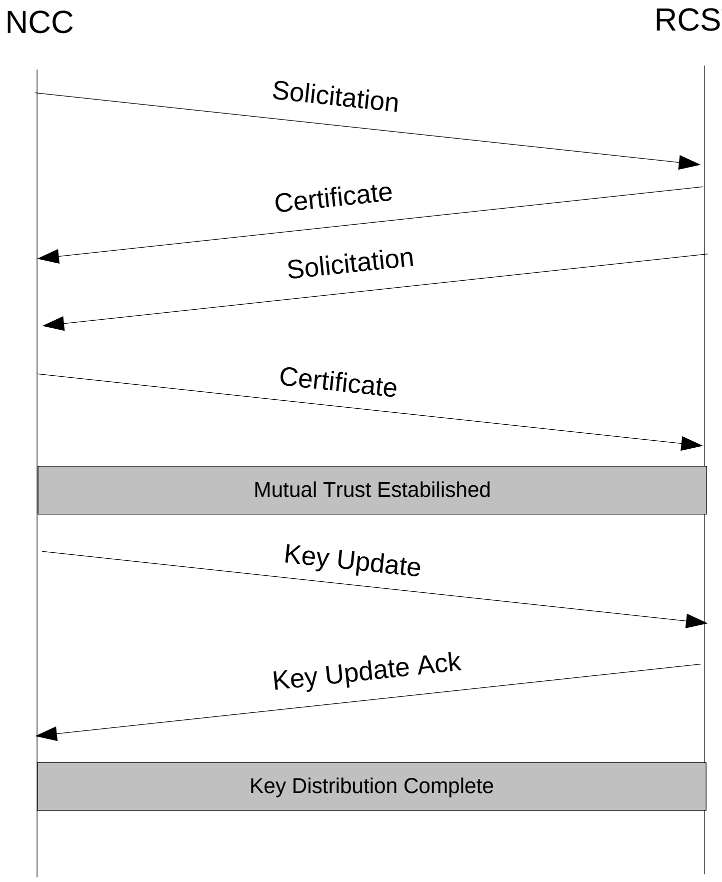


Figure 8.8: Key distribution protocol

8.4.3.2.2 Key Update and Rollover

A peer may initiate a key update in an unsolicited fashion as needed. The data structure used to complete a key update is illustrated in Figure 8.9.

Fallow	00	<Key>
Fallow	01	<Key>
Current	10	<Key>
Next	11	<Key>

Figure 8.9: Key ring

This data structure conceptually consists of a set of pointers (Current, Next, Fallow), a 2 bit identification field (utilized in the Encryption Headers as described in clause 8.4.4), and the actual symmetric keys themselves. A key update consists of generating a new key, placing it in the last fallow slot just prior to the Current pointer, updating the next pointers and current pointers in a circular fashion, and generating a Key Update message reflecting these changes. The key update mechanism allows for multiple keys to be "in play" simultaneously, so that seamless key rollovers can be achieved. A seamless key rollover is ensured by always transmitting the 2-bit identification field ("key ring position") which identifies the key used for encryption.

The NCC can carry out the key rollover at any time after the key update has been completed, simply by starting to use the "next" key. In the inbound, the RCST carries out the key rollover as soon as a key rollover in the forward link has been detected. It is the responsibility of the NCC to ensure that the appropriate keys are available before carrying out a key rollover.

When the present TRANSEC implementation is employed, this key rollover mechanism is mandated for maintenance of the ACC and DCC keys in the RCST.

8.4.3.2.3 Host Keying

The host keying protocol is defined in Figure 8.10. This protocol defines how any host is originally provided an X.509 certificate from a Certificate Authority. The Certificate Authority Foundry can be part of the Security management System in the NCC or can be a remote entity accessed through a secure protocol.

The messages that constitute this protocol are defined in X.509 [i.24]. The messages are exchanged using TCP.

Following completion of the initial certificate exchange shown in Figure 8.9, the hub will transmit the network acquisition key to the host. The network acquisition key is encrypted with the host public key before being transmitted to the host.

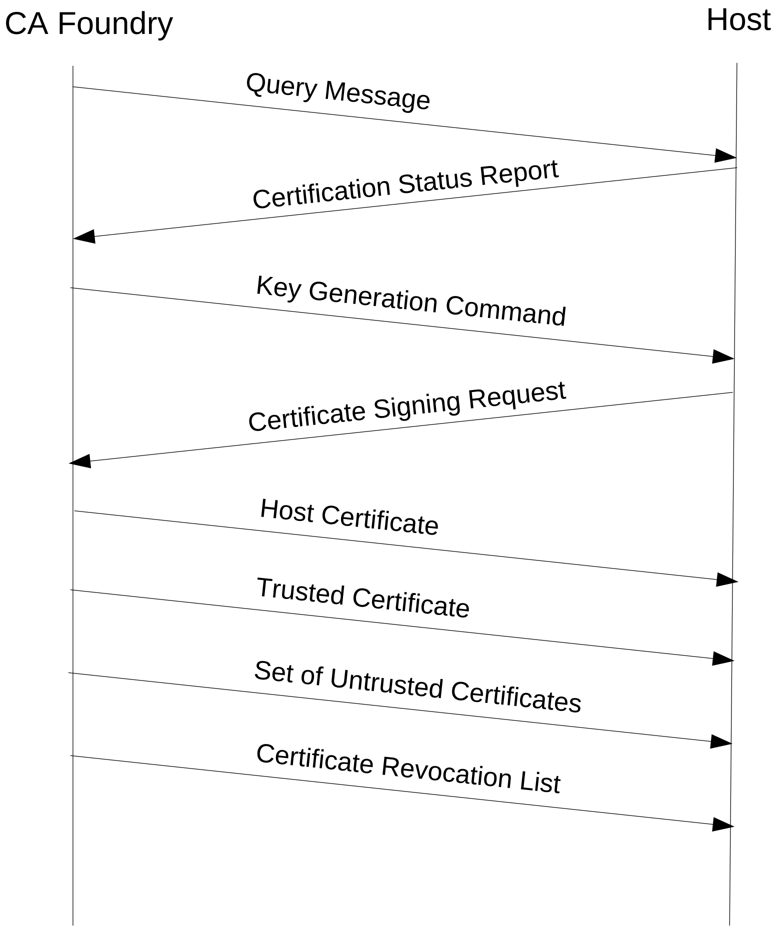

CA Foundry
Host
Query Message
Certification Status Report
Key Generation Command
Certificate Signing Request
Host Certificate
Trusted Certificate
Set of Untrusted Certificates
Certificate Revocation List

Figure 8.10: Host keying protocol

8.4.3.3 Key Management for Acquisition Ciphertext Channel

The ACC key is provided by the NCC. It is never exposed in the clear. When transferred, it is always encrypted with the host's public key.

NOTE: Because an RCST cannot enter the network without the current ACC key, an RCST that is out of the network for the entire period between the time a new key is pushed and the time it is activated will be unable to join the network until a new key is entered by other means, as defined in clause 8.4.6.2. Because of this, the operator should select a crypto-period which balances the operational needs of the network against preserving the theoretical strength of the ACC key.

The key roll operates in a manner similar to the dynamic key rollover.

1) When an RCST first enters the network, it is given the "current" ACC Key and the "next" ACC key. The "next" key is the one that will be used after the next key roll.

2) In the return direction, RCST carries out a key rollover when one is detected in the forward link.

3) When the key rollover occurs, the remote uses the "next" ACC KEY.

4) After the key rollover, a new "next" key is generated by the NCC.

5) The new current/next key pair is pushed to all RCST's.

8.4.3.4 Authentication and Key Management Protocol Messages

This clause defines the method for exchanging key management protocol messages and the contents of these messages.

8.4.3.4.1 Certificate Solicitation Message

The format of the certificate solicitation message is shown in Table 8.9.

Table 8.9: Syntax of certificate solicitation message

Syntax	No. of bits		Information Mnemonic
	Reserved	Information	
certificate_solicitation_message() {			
RCST_MAC_address		48	uimsbf
interactive_network_ID		16	uimsbf
rx_session_id		32	uimsbf
peer_tx_session_id		32	uimsbf
tx_session_id		32	uimsbf
peer_rx_session_id		32	uimsbf
rx_session_state	3	2	uimsbf
tx_session_state		2	uimsbf
rx_certification_state		1	bslbf
rx_certificate_id		32	see text
tx_certificate_id		32	see text
algorithm_length	4	4	uimsbf
for (i=0; i< algorithm_length; i++) {			
Algorithm		16	uimsbf
}			
transport_set_count		8	uimsbf
for (i=0; i< transport_set_count; i++) {			
transport_set_start		4	uimsbf
transport_set_end		4	uimsbf
}			
message_length		8	uimsbf
for (i=0; i< message_length; i++) {			
message_byte		8	
}			
}			

Semantics for certificate_solicitation_message:

RCST_MAC_address: RCST hardware address as per IEEE 802.3 [i.26].

interactive_network_id: This 16 bit field gives the label identifying the network_ID for the interactive network.

rx_session_id: This field defines the transmitting node's ID for incoming exchanges. The initial value is set to a random number when the node is initialized. Thereafter the value is incremented once every constant time period (e.g. every fifteen minutes). The time period is configured during system installation.

peer_tx_session_id: If there have been no exchanges from the peer receiving this message, then this field has a value of zero. Otherwise the value of the tx_session_id most recently received from the peer is copied into this field.

tx_session_id: This field defines the transmitting node's ID for outgoing exchanges. The initial value is set to a random number when the node is initialized. Thereafter the value is incremented once for each Key Update.

NOTE: If the link goes down and the RCST needs to re-acquire, there will be a Key Update as part of that activity; thus this value will also be incremented in that case.

peer_rx_session_id: If there have been no exchanges from the peer receiving this message, then this field has a value of zero. Otherwise the value of the rx_session_id most recently received from the peer is copied into this field.

rx_session_state: This field defines the status of the session from the receiving peer to the transmitting node. The allowed values are defined in Table 8.10.

Table 8.10: Values for rx_session_state and tx_session_state

Value	State
0	Session up
1	Session down
2	Session opening
3	Reserved

tx_session_state: This field defines the status of the session from the transmitting to the receiving node of the present message. The allowed values and their interpretation are the same as for rx_session_state (Table 8.10).

rx_certification_state: This field defines the status of a certificate received from the peer receiving this message. It takes the value '1' if a certificate has been received and '0' otherwise.

rx_certificate_id: If there have been no exchanges from the peer receiving this message, then this field has a value of zero. Otherwise it is set as follows: The 20-byte ASN.1 digest [i.27] of the peer's certificate is computed. The first four bytes (0..3) of this are placed into the field, with byte 0 being placed in the least significant eight bits, through byte 3 being placed in the most significant eight bits.

tx_certificate_id: This field is set as follows: The 20-byte ASN.1 digest [i.27] of the node's certificate is computed. The first four bytes (0..3) of this are placed into the field, with byte 0 being placed in the least significant eight bits, through byte 3 being placed in the most significant eight bits.

algorithm_length: This field defines the number of encryption algorithms that the transmitting node is capable of using.

algorithm: This field contains the identifying number of an algorithm which the node is capable of using. The allowed values are defined in Table 8.11.

Table 8.11: Encryption algorithm identifying numbers

Algorithm	Number
RSA-2048	500
AES-256 CBC	507
AES-256 CFB	508
AES-256 CTR	509

NOTE 1: The AES-256 algorithm is also used in ECB mode, but only for computation of initialization vectors, not for actual data encryption.

NOTE 2: The transport_set_count, transport_set_start and transport_set_end fields are included for future enhancement. There is currently one transport_set defined, with fixed values. In general, transport_sets define the keys usable for TRANSEC certificates.

transport_set_count: This field is set to 1.

transport_set_start: This field is set to 2.

transport_set_end: This field is set to 3.

message_length: This field is set to the length, in bytes, of the optional text message field. if no message is included, this field is set to 0.

message_byte: One byte of an optional text message field. The presence, contents and usage of this message are all unspecified.

In the forward link, the solicitation message is transported in the TIM in a TRANSEC_ message_descriptor with a transec_message_type value of 0x02. The transec_message_byte sequence contains the certificate_sollicitation_message formatted according to Table 8.9.

In the return link, the message is transported in a PDU using the TRANSEC_certificate protocol type. The format of the PDU is as specified in Table 8.12. The message_type field is set to the value 0x02 for a certificate_solicitation message.

Table 8.12: PDU format for certificate solicitation messages

Syntax	No. of bits	Mnemonic
Certificate_message(){		
message_type	8	uimsbf
certificate_solicitation_message	see text	
}		

8.4.3.4.2 Certificate Presentation Message

The certificate is formatted in accordance with the X.509 standard [i.24]. In the forward link, it is transported in the TIM, in a TRANSEC_message_descriptor with a transec_message_type value of 0x03. The transec_message_byte sequence contains the certificate, formatted according to [i.24].

In the return link, the certificate is transported in a PDU using the TRANSEC_certificate protocol type. The format of the PDU is as specified in Table 8.13. The message_type field is set to the value 0x03 for a certificate presentation message.

Table 8.13: PDU format for certificate presentation messages

Syntax	No. of bits	Mnemonic
Certificate_message(){		
message_type	8	uimsbf
Certificate	As per [i.24]	
}		

8.4.3.4.3 Key Update Message

In the forward link, they key update message is transported in the TIM, in a TRANSEC_message_descriptor with a transec_message_type value of 0x03. The transec_message_byte sequence contains the key update message, formatted according to Table 8.14.

Table 8.14: Syntax of key_update_message

Syntax	No. of bits		Information Mnemonic
	Reserved	Information	
key_update_message() {			
message_type		16	uimsbf
payload_length		8	uimsbf
for (i=0; i< payload_length; i++) {			
payload_byte		8	see text
}			
algorithm_length	3	5	uimsbf
for (i=0; i<algorithm_length; i++) {			
algorithm_byte		8	see text
}			
for (i=0; i< AF; i++) {		8	
filler_byte		8	uimsbf
}			
signature_length		8	uimsbf
for (i=0; i< signature_length; i++) {			
signature_byte		8	see text
}			
interactive_network_id		16	uimsbf
}			

Semantics for key_update_message:

message_type: This field identifies the message type. It is set to "1012" (decimal).

payload_length: This field defines the number of bytes contained in the payload field.

payload_byte: Each instance of this field contains one byte of the encrypted contents of the Key Update Command. Each instance of the field has been encrypted first with the RCST's public RSA key, and secondly with the NCC's private RSA key. The format of the overall payload field is defined in clause 8.4.3.4.3.1.

algorithm_length: This field defines the number of bytes in the algorithm field.

algorithm_byte: Each instance of this field contains one byte of the name of the encryption algorithm used. Valid names are listed in Table 8.15.

Table 8.15: Permitted encryption algorithm names

Algorithm
"ALG_AES_CBC"
"ALG_AES_CFB"
"ALG_AES_CTR"

filler_byte: Unused byte; the number AF of these bytes is such that AF+algorithm_length is an integer multiple of 4.

signature_length: This field defines the number of bytes in the signature field.

signature_byte: Each instance of this field contains one byte of the signature field. The signature consists of the first twenty bytes of the new key, encrypted first by the destination RCST's Public RSA key, and secondly by the NCC's Private RSA key.

interactive_network_id: This 16 bit field gives the label identifying the network_ID for the interactive network.

8.4.3.4.3.1 Key Update Payload Format

The cleartext of the payload field is formatted as defined in Table 8.16.

Table 8.16: Syntax of key update payload field

Syntax	No. of bits		Information Mnemonic
	Reserved	Information	
key_update_payload() {			
tx_session_id		32	uimsbf
rx_session_id		32	uimsbf
command_count		8	uimsbf
for (i=0; i< command_count; i++) {			
transport_id		8	uimsbf
active_key_index	6	2	uimsbf
new_key_count		8	uimsbf
for (j=0; j< new_key_count; j++) {			
key_ring_index	5	2	uimsbf
valid		1	bslbf
key_length		8	uimsbf
for (k=0; k<key_length; k++) {			
key_byte		8	see text
}			
}			
}			
}			

Semantics for key_update_payload:

tx_session_id: The semantics for this field are the same as for the synonymous field in the certificate_solicitation_message (clause 8.3.4.1);

rx_session_id: The semantics for this field are the same as for the synonymous field in the certificate_solicitation_message (clause 8.3.4.1);

command_count: This field defines the number of key update commands that follow;

transport_id: This field identifies the purpose of the command. Allowed values are 2 for "TRANSEC Master TX Netkey" (hub-to-remote) and 3 for "TRANSEC master RX Netkey" (remote-to-hub);

active_key_index: Identifies the key ring position of the currently active key;

new_key_count: This field identifies the number of new keys that follow;

key_ring_index: The index to the RCST's key ring position where the key is to be stored;

valid: Validity of the new key. This field should always be set to '1';

key_length: This field identifies the length of the key in bytes;

key_byte: Each instance of this field contains one byte of the actual key.

8.4.3.4.4 Key Update Acknowledgement

This message is sent only in the return link. The format of this message is identical to the Certificate solicitation message (clause 8.3.4.1). If the message is being used as a Certificate Solicitation message, then either the peer_tx_session_id, the peer_rx_session_id, or both, will be zero. This indicates that a Certificate has not been received and successfully processed. If both of these fields do have values, and if these values match what the NCC expects to see from this RCST, then the message is a Key Update Acknowledgement.

8.4.4 Encryption

8.4.4.1 Forward Link

Each DVB-S2 frame in the forward link is encrypted using AES-256 in CBC mode, as defined in this clause. The CBC operation is defined in clause 8.4.3.1.

The organization of a BBFRAME with TRANSEC is illustrated in Figure 8.11.

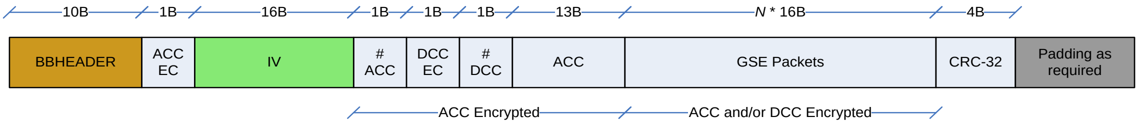


Figure 8.11: DVB-S2 BBFRAME with TRANSEC Encrypted GSE Packet(s)

The fields in this frame are as follows:

- The BBHEADER is as per the DVB-S2 standard and is sent unencrypted. The DFL indicates the payload length from the start of the ACC EC field to the end of the CRC-32. The SYNC field within the BBHEADER is set to indicate the use of Generic Stream Encapsulation with CRC, as per [i.28].
- The ACC EC field is sent unencrypted. It is a one-byte field that describes the ACC encryption in the current frame. The format of this field is defined in Table 8.17.

Table 8.17: ACC EC field

Bits	Description
7 (MSB)	IV presence indicator. Should be set to 1.
6:2	Reserved (set to 1)
1:0	ACC key index

- The 16-byte Initialization vector (IV) is sent unencrypted. At system start-up, the initial IV should be the result of one of the Known Answer Tests defined for the AES encryption algorithm. For each subsequent BBFRAME, the last 16 bytes of encrypted data are used as the initial IV for the following BBFRAME.

NOTE: Since an initial IV is transmitted in each BBFRAME, this method works even in situations where only some frames can be demodulated by a particular RCST; for example in ACM systems.

- The #ACC field is a one-byte field. It indicates the number of 16-byte chunks that are encrypted using the ACC key. Since the first chunk is always ACC encrypted, this number is always at least equal to 1. It is itself encrypted using the ACC key, as described below.
- The DCC EC field is a one-byte field that describes the DCC encryption in the current frame. The format of this field is defined in Table 8.18. It is encrypted using the ACC key, as described below.

Table 8.18: DCC EC field

Bits	Description
7 (MSB)	IV presence indicator. Should be set to 1.
6:2	Reserved (set to 1)
1:0	DCC key index

- The #DCC field is a one-byte field. It indicates the number of 16-byte chunks that are encrypted using the DCC key. It is itself encrypted using the ACC key, as described below.
- The GSE packets constitute the actual information payload of the frame. They are encrypted using the ACC and DCC keys as described below.
- The CRC-32 is a 4-byte field. It is computed over the frame content in the same manner as for non-TRANSEC implementations, in accordance with [i.28].
- Padding is appended as required to fill the BBFRAME, as per the DVB-S2 standard.

The complete, encrypted BBFRAME is forwarded to the mode adaptation interface of the DVB-S2 modulator.

8.4.4.1.1 Encryption Process

Each frame contains at least one chunk encrypted using the ACC key. As illustrated in Figure 6, this chunk contains the #ACC, DCC EC and #DCC fields, plus the first 13 bytes of actual ACC data.

This is followed by #ACC-1 further chunks of ACC-encrypted data and #DCC chunks of DCC-encrypted data. The Cipher Block Chaining process is carried over from the ACC to the DCC, as illustrated in Figure 8.12.

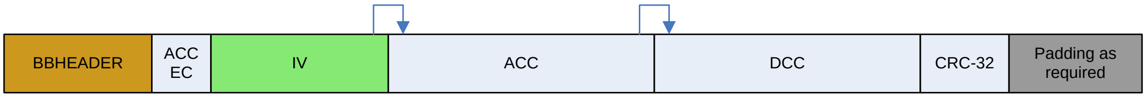


Figure 8.12: Cipher block chaining process

The ACC and DCC each consist of an integer number of 16-byte chunks. Unused space at the end of each channel should be filled by a chaff packet. If the amount of unused space is insufficient for the insertion of a chaff packet, it should be filled with pseudo-random data from the Chaff Packet generator's RNG (clause 8.4.5.6). It is the responsibility of the receiver to detect this condition and discard the excess bytes.

8.4.4.1.2 Decryption Process

The decryption process is the inverse of the encryption. It proceeds as follows:

- Physical-layer recovery of the frame is performed, including the CRC-32 check and stream filtering to discard irrelevant frames.
- The IV is loaded into the AES engine and the first chunk is decrypted using the ACC key indicated in the ACC EC field.
- The number of subsequent ACC chunks and the total number of DCC chunks are determined from the decrypted fields.
- Remaining ACC chunks are decrypted; chaff packets and any trailing random data are discarded and the recovered GSE packets forwarded to the de-capsulation.

- DCC chunks are decrypted; chaff packets and any trailing random data are discarded and the recovered GSE packets forwarded to the de-capsulation.

8.4.4.2 Return Link and Mesh Transmissions

Each burst in the return link and mesh links are encrypted using AES-256 in CFB mode, as described in this clause. The CFB mode is defined in clause 8.4.4.3.2.

Each burst serves only the ACC or DCC. The encryption process ensures that the actual channel used can be inferred from the encrypted burst only by the intended recipient.

The organization of a TRANSEC burst payload is illustrated in Figure 8.13.

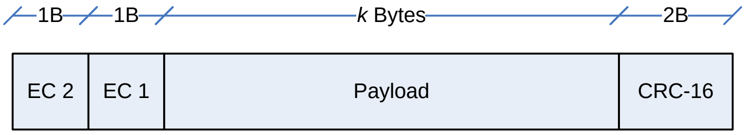


Figure 8.13: Burst with TRANSEC encryption

The fields in this are as follows:

- The EC2 field is sent unencrypted. It is a one-byte field which describes the key used to encrypt the EC1 field. The format of this field is defined in Table 8.19.

Table 8.19: EC2 field

Bits	Description
7:3 (MSB)	Reserved (set to 1)
2	Key selector. Should be set to 1 to denote ACC.
1:0	Key index

- The EC1 field is a one-byte field which describes the key used to encrypt the payload and CRC-16. The format of this field is defined in Table 8.20.

Table 8.20: EC1 field

Bits	Description
7:3 (MSB)	Reserved (set to 1)
2	Key selector. (0=DCC, 1=ACC).
1:0	Key index

- The payload field constitutes the actual information content of the burst. It is encrypted using the ACC or DCC key as indicated by the EC1 field. For TRANSEC purposes, the number of bytes k in this field can be any value greater than or equal to 29.
- The CRC-16 is a two-byte field. It is computed over the entire, unencrypted and unscrambled content of the burst, from the EC2 field to the end of the payload. When this CRC is used, no other burst CRC should be used. The polynomial for this CRC is $1 + x^5 + x^{12} + x^{16}$.

The complete, encrypted burst payload is forwarded to the scrambling, FEC encoding and modulation function.

8.4.4.2.1 Encryption Process

The encryption process consists of the following steps, as illustrated in Figure 8.14:

- The CRC-16 is computed.
- The k+2 bytes of the payload and CRC-16 are encrypted using CFB mode; using the appropriate key as identified in the EC1 field and an initial IV computed according to the provisions in clause 8.4.4.2.3.

- The EC1 field and the first 15 bytes of the encrypted payload-plus-CRC are encrypted using CBC mode; using the ACC key identified in the EC2 field and using as IV the next 16 bytes (bytes 16 - 31) of the ciphertext resulting from the encryption of the payload and CRC-16.

- The EC2 field is transmitted in cleartext.

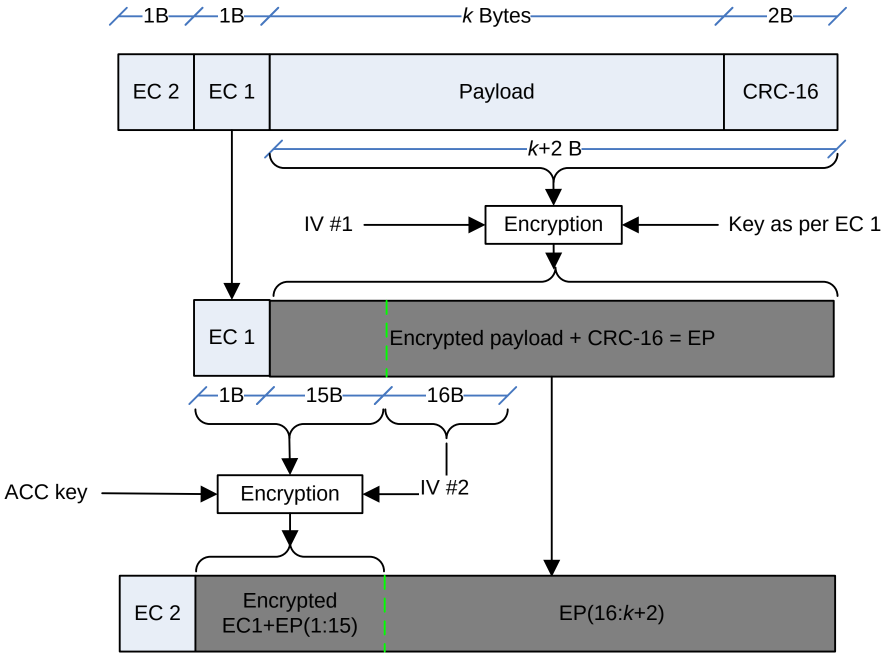


Figure 8.14: Burst Encryption Process

8.4.4.2.2 Decryption Process

The decryption process is the inverse of the encryption. It proceeds as follows:

- Physical-layer recovery of the burst is performed.

- The EC1 field and the first 15 bytes of the encrypted payload-plus-CRC are decrypted using the ACC key identified in the EC2 field; using as IV the next 16 bytes of the encrypted payload-plus-CRC.

- The key to be used for decryption of the payload is determined from the EC1 field.

- The payload and CRC-16 are decrypted using the key identified in the EC1 field and an initial IV computed according to the provisions in clause 8.4.4.2.3.

- The CRC-16 is checked. This not only provides payload integrity, but also ensures that consistent keys have been used in the encryption and decryption.

8.4.4.2.3 initialization Vectors for Burst Transmission

Initialization Vectors for encryption of the first block of each burst transmission are not transmitted over the air, but are generated at both ends of the link in a synchronized manner. The IV is obtained by encryption of a 128-bit string, using AES-256 in ECB mode and the current ACC key. The composition of the 128-bit string is shown in Table 8.21.

Table 8.21: Burst IV plaintext composition

Bits	Description
69 (MSB)	Random number
8	Superframe_sequence
16	Superframe_Count_Extension
16	Superframe_Count
8	frame_number
11 (LSB)	Timeslot_Number

Values for these fields are obtained as follows:

The random number is generated by the NCC and communicated periodically to the RCST's using the TRANSEC_message_descriptor. This version of the TRANSEC_message_descriptor is sent at regular intervals in the TIM; it includes identification of the superframes to which it applies:

- The superframe_sequence is the normal DVB-RCS2 identifier of the superframe sequence to which the RCST is logged in.
- The 16-bit Superframe_Count_Extension is an extension of the conventional Superframe_Count. Its value is communicated periodically to the RCST's using the TRANSEC_message_descriptor in the TIM. Once set, the RCST increments the Superframe_Count_Extension whenever the regular Superframe_Count rolls over from 65 535 to 0.
- Superframe_Count, frame_number and Timeslot_Number are the conventional DVB-RCS2 identifiers of the time slot in which the transmission takes place.

In order to avoid re-setting, the Superframe_Count_Extension should be set at system re-boot to the number of Superframe_Count wrap-around periods elapsed since January 1, 1970 00:00UTC, modulo 65 536.

The NCC may renew the value of the random number as desired. The new value is communicated using the TRANSEC_message_descriptor with a transec_message_type value of 0x01 and including the random number. The transec_message_byte sequence is formatted in accordance with Table 8.22.

Table 8.22: Return / mesh link IV descriptor sub-type

Syntax	No. of bits		Information Mnemonic
	Reserved (see note)	Information	
return_link_IV_update(){			
superframe_loop_count		8	Uimsbf
for (i=0; i<= superframe_loop_count; i++) {			
random_present	7	1	
if (random_present ==1) {			
random_number	3	69	Uimsbf
}			
superframe_sequence		8	Uimsbf
superframe_count_extension		16	Uimsbf
superframe_count		16	Uimsbf
}			Uimsbf
}			

The semantics for the return_link_IV_update are as follows:

superframe_loop_count: This is an 8-bit field indicating one less than the number of iterations of the loop that follows. A zero count indicates one iteration.

random_present: This flag indicates the presence or absence of the random_number field. The value is set to 1.

random_number: This is a 72-bit field that contains bits that may be used as the MSB's of the IV for the first block of each burst payload. The value applies to the superframe identified by superframe_ID and takes effect from the superframe identified by the superframe_count_extension and superframe_count. If that time is in the past, the value takes effect immediately.

superframe_sequence: This 8-bit field identifies the superframe sequence to which the parameters apply.

superframe_count_extension: This 16-bit field defines the value of superframe_count_extension in force in combination with superframe_count. The RCST increments its local value of superframe_count_extension whenever superframe_count rolls over from 65535 to 0.

superframe_count: This 16-bit field identifies the modulo-65536 superframe count to which the descriptor entry applies.

8.4.4.3 AES-256 Operational Modes

This clause defines the block cipher operation modes in which the AES-256 algorithm is used in the system. Figure 8.15 to Figure 8.19 are obtained from Wikipedia.org and are placed in the public domain by their author by the statement: "This image has been released into the public domain by its author, Lunkwill. This applies worldwide."

8.4.4.3.1 Cipher Block Chaining (CBC) Mode

CBC is the preferred operational mode from a security perspective and is the mode applied in the forward link. CBC encryption is illustrated in Figure 8.15. Each block of 256 bits of plaintext is excusive-or'ed with the initialization Vector (IV). The encryption algorithm is applied to the result, using the appropriate key. The result of this is the ciphertext. The first block in a frame uses the IV supplied in the frame. The ciphertext resulting from each encryption operation is used as IV for the encryption of the subsequent block.

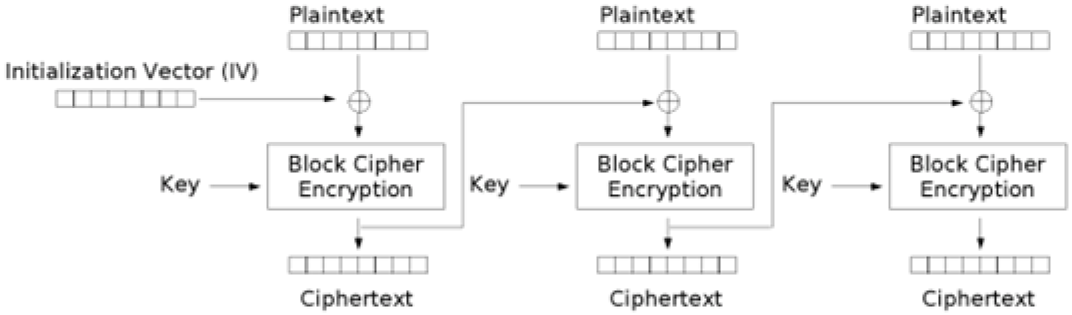


Figure 8.15: CBC Encryption

The corresponding decryption is shown in Figure 8.16. The decryption algorithm is applied to each 256-bit block of ciphertext, using the appropriate key. The result is exclusive-or'ed with the IV to produce the plaintext block. The first block in a frame uses the IV supplied in the frame. Each subsequent block uses the ciphertext of the preceding block as IV.

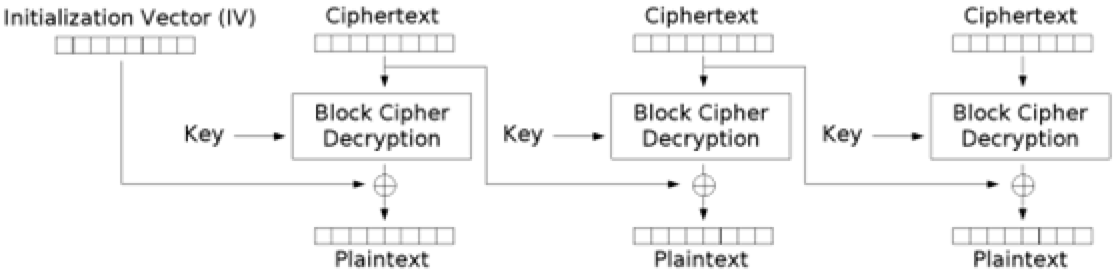


Figure 8.16: CBC Decryption

8.4.4.3.2 Cipher Feedback (CFB) Mode

This mode is also acceptable from a security perspective; it is applied for burst transmission in the return link and for mesh.

Encryption is shown in Figure 8.17. The initialization Vector (IV) is encrypted using the AES-256 algorithm, using the appropriate key. The result is exclusive-or'ed with the plaintext to form the ciphertext. The first block in a frame uses the IV supplied externally. The ciphertext resulting from each encryption operation is used as IV for the encryption of the subsequent block.

Since the last block of ciphertext is not used as a subsequent IV, it is acceptable for this to be shorter than 256 bits. Such a shortened ciphertext block is produced by exclusive-or'ing the initial output bits from the encryption operation with corresponding plaintext bits.

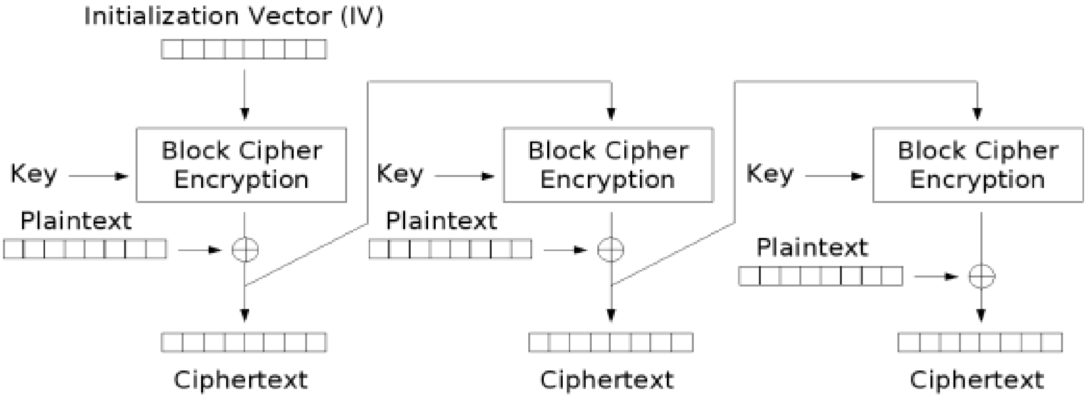


Figure 8.17: CFB encryption

The corresponding decryption is shown in Figure 8.18. The initialization Vector (IV) is encrypted by the AES-256 algorithm, using the appropriate key. The result is exclusive-or'ed with the ciphertext to form the plaintext. The first block in a frame uses the IV supplied externally. Each block of ciphertext also serves as IV for decryption of the subsequent block.

Since the last block of ciphertext is not used as a subsequent IV, it is acceptable for this to be shorter than 256 bits. In this case, the plaintext block is produced by exclusive-or'ing the initial output bits from the encryption operation with corresponding ciphertext bits.

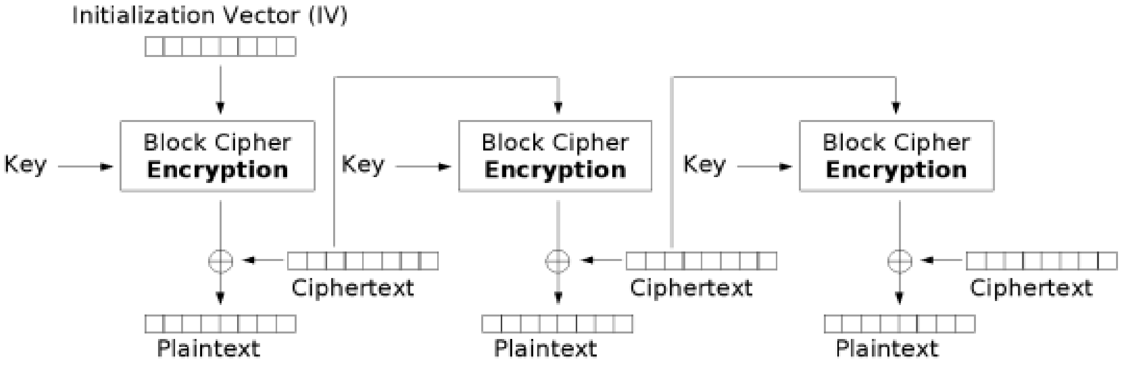


Figure 8.18: CFB decryption

8.4.4.3.3 Electronic Code Book (ECB) Mode

This mode is used only to randomize vectors internally in equipment. Encryption is shown in Figure 8.19. Each block of plaintext is encrypted individually, using the appropriate key. ECB decryption is not used in this application.

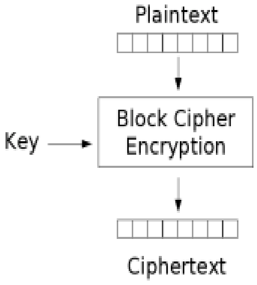


Figure 8.19: ECB encryption

8.4.4.4 Public-Key Encryption

Public-key encryption and decryption are carried out using RSA-2048 [i.25].

8.4.5 Traffic Activity Obfuscation

8.4.5.1 Forward Link Chaff Packets and Fill Frames

Any part of a forward link frame that cannot be filled with actual data is filled with chaff packets. This packet is an un-fragmented GSE packet transmitted in broadcast mode; i.e. with no address label and using the TRANSEC_chaff protocol type. The packet payload is random data. Use of the random number generator specified in clause 8.4.5.6 is recommended.

If the part of the frame to be filled is too small for a complete GSE packet with minimal header, it should be filled with random data.

DVB-S2 dummy frames are not transmitted. If there is no data to transmit in a frame, is filled entirely with chaff packets. These can be transmitted in the ACC and DCC as desired, following the normal rules.

8.4.5.2 Dummy Demand-Assigned Traffic Bursts

All demand-assigned traffic slots in the MF-TDMA frame are assigned to RCST's unless prevented by fundamental hardware constraints such as the single-carrier nature of the transmitter. An RCST always transmits in an assigned traffic slot. If the RCST has no useful data to transmit, the burst payload is filled with random "chaff" data. Use of the random number generator specified in clause 8.4.5.6 is recommended. The random payload is encrypted using the DCC key.

The slot assignments are made in such a way that the long-term average of the number of bursts transmitted by all RCST's per unit of time is approximately equal, while respecting the actual capacity demands of the individual RCST's.

8.4.5.3 Logon Burst Assignment and Randomization

The use of assigned logon transmissions is preferred to contention-based transmission. The procedure defined in this clause is used to obfuscate the actual logon activity in the network.

Each logon slot in the superframe is used in one of three ways: As a dummy slot assigned to an already-logged on RCST, as an actual logon slot, or unused.

A logged-on RCST always transmits in an assigned logon slot. The payload of the burst is random "chaff" data. Use of the random number generator specified in clause 8.4.5.6 is recommended. The random payload is encrypted using the ACC key. The transmit power is the same as that used for actual logon transmissions. The burst transmit instant is chosen randomly in such a manner that the burst arrival time is uniformly distributed over the logon slot at the NCC, but contained entirely within this slot.

The burst transmit frequency is offset from the nominal value by a random quantity, chosen uniformly in an interval corresponding to the system's frequency tolerance for logon bursts.

An RCST that is not logged on transmits a conventional logon burst in an assigned slot. No transmissions take place in unused slots.

The use of each logon slot in each superframe is chosen randomly, with probabilities as given in Table 8.23. Use of the random number generator specified in clause 8.4.5.6 is recommended.

Table 8.23: Logon slot assignment probabilities

Slot usage	Probability
Dummy	$P_1 - M / N$
Actual	P_2
Unused	$(1 - P_1 - P_2) + M / N$

Each time an actual logon burst is detected, the NCC starts a counter which is initialized to N and is decremented for each superframe by the total number of logon slots in the superframe. The counter is deactivated when it reaches 0. M is the number of active counters in any given superframe.

The probabilities P_1 and P_2 are system design parameters, chosen such that $P_1 + P_2 < 1$.

NOTE: The choice of P_1 and P_2 and *N* are a trade-off between on the one hand the degree to which the actual activity can be hidden and on the other hand the amount of capacity set aside for this purpose. Typical values are $P_1 = 0,5$, $P_2 = 0,25$ and *N* equal to the total number of logon slots in 100 seconds.

Assignment of dummy slots to the logged-on RCST population are randomized but such that the average number of dummy transmissions is approximately equal for all RCST's.

8.4.5.4 Control Burst Transmissions

The use of assigned control burst transmissions is preferred to contention-based transmission. All control slots in the MF-TDMA frame should be assigned to RCST's unless prohibited by hardware constraints such as the single-carrier nature of the transmitter. An RCST always transmits in an assigned control slot.

The control slot assignments are made in such a way that the average interval between the corresponding transmissions is approximately equal for all active RCST's. The slot positions for individual RCST's are randomized. Use of the random number generator specified in clause 8.4.5.6 is recommended. It is recommended to assign the slots through the TBTP2, rather than through the control assign descriptor.

Power setting, timing offset and frequency offset of dummy control transmissions are made in the same manner as that defined for logon transmissions in clause 8.4.5.3.

8.4.5.5 Contention-based Control and Logon Transmission

Contention-based transmission of traffic slots should not be used. The use of contention-based transmission logon and control burst transmissions is not recommended. If this method is nevertheless used, the process defined in this clause can be used to obfuscate the actual activity. For the purpose of this clause, the term "slot" means a logon or control slot used for contention-based access.

Each slot is used in one of three ways: As a dummy slot assigned to an already-logged on RCST, as an actual contention slot, or unused.

A logged-on RCST always transmits in an assigned slot. A control burst transmission is in accordance with the normal rules for assigned control slots. A logon transmission made by a logged-on RCST in an assigned slot is in accordance with the corresponding provisions of clause 8.4.5.3, including the randomization of timing and frequency.

Actual contention slots can be used by RCST's for contention access in accordance with the normal rules. Unused slots are identified in the TBTP2 by assigning them to a non-existent RCST.

The use of each contention slot in each superframe should be chosen randomly, using the probability distribution defined in clause 8.4.5.3. Separate counter sets should be maintained for control and logon slots.

Assignment of dummy contention slots to the logged-on RCST population should be randomized but such that the average number of dummy transmissions is approximately equal for all RCST's.

8.4.5.6 Random Number Generation

The following method (from [i.29]) is recommended for generation of pseudo-random data.

The algorithm has 4 state variables: *x, y, z, w* (all are 32-bit unsigned values).

The state variables are initialized to *x* = 123456789; *y* = 362436069; *z* = 521288629; *w* = 88675123. Initialization should take place at system re-boot only. Following initialization, the random number generator is exercised a random number of times and the corresponding values discarded. The number should not be less than 1,000 and be derived from the date and time obtained from an NTP server.

Computation of the next pseudo-random number w is carried out as follows:

```
t = x ^ (x << 11);

x = y;

y = z;

z = w;

w = (w ^ (w >> 19)) ^ (t ^ (t >> 8));
```

Where "^" denotes bit-wise exclusive-or (modulo-2 addition), "*a* << *b*" indicates *a* left-shifted by *b* bit positions and "*a* >> *b*" indicates *a* right-shifted by *b* bit positions.

8.4.6 Special Considerations

8.4.6.1 Beam Handover in Mobile Systems

In order to facilitate beam handover and initial entry of mobile terminals in a TRANSEC-secured system that employs multiple beams and even multiple satellites and gateways, it is necessary that the same ACC keys are used in all forward links. One security management system will be responsible for generating these keys and for initiating simultaneous roll-overs. Responsibility for this function may be with the security management system of one of the networks, or with an external entity. The keys and roll-over commands are communicated to the security management systems of the affected networks using a secure link.

8.4.6.2 Manual Key Entry

For initial commissioning, it should also be possible to enter a network acquisition key manually, before the RCST has generated an X.509 certificate. The key is protected by a user-generated passphrase of any desired length. Transmitting the key to the RCST is done outside of the system, either verbally (e.g. over a phone line) or in a file transfer. This procedure can also be used to bring in an RCST which has missed the ACC key rollover twice.

The procedure is as follows:

1) The operator of the Security Management System enters a key generation command which includes the desired passphrase and the hardware MAC address of the RCST. The passphrase is a UTF-8 [i.30] string of any length.

2) The Security Management System takes the RCST hardware address (48-bit IEEE MAC address as per [i.5]), passphrase, and appends a 64 bit random number ("salt") to form a string. This is turned into a 256 bit key using PBKDF2 as described in [i.31]. The iteration count should be 1 000.

3) The key generated in the previous step is used to encrypt the ACC key using AES-256 in ECB mode.

   The 310 bits result, consisting of 256 encrypted key bits + 64 salt bits are transmitted to the operator of the RCST. They are encoded for ease of transmission as follows. For this process, the term "checksum" means SHA-512 hash [i.32] truncated to use only the bits at the beginning.

4) The Security Management System presents three strings which contains information to be transmitted to the remote. The three strings can be encoded using either base-32 or base-64 encoding in accordance with [i.33]. The strings are constructed as illustrated in Figure 8.20 and described in the following:

1) String 1 consists of the following elements:

   i) A five-bit header, which is formatted as follows:

      1) One reserved bit.

      2) Two bits indicating the key ring position of the key.

      3) One reserved bit.

      4) One bit indicating the main encoding scheme used ('0' indicates base-64, '1' indicates base-32).

         This header is itself base-32 encoded to occupy one byte.

   ii) The first 14 bytes (1 - 14) of the AES-encrypted ACC key.

   iii) Chsum1, which is a 2 byte checksum computed over the preceding 15 bytes.

   The concatenation of (ii) and (iii) are encoded according to the selected scheme. The complete string contains either 23 or 27 bytes, depending on the encoding.

2) String 2 consists of the following elements:

   i) 15 bytes (15-29) of the AES-encrypted ACC key.

ii) Chsum2, which is a 2 byte checksum computed over the preceding 15 bytes.

The concatenation of (i) and (ii) are encoded according to the selected scheme. The complete string contains either 23 or 28 bytes, depending on the encoding.

3) String 3 consists of the following elements:

i) The last 3 bytes of the AES-encrypted ACC KEY.

ii) The 8 bytes "salt".

iii) Chsum 3, which is a 2 byte checksum computed over the concatenation of the entire encrypted ACC key and the "salt".

iv) Chsum4, which is a 2 byte checksum computed over the concatenation of the entire encrypted ACC key, the "salt", the passphrase and the RCST's MAC address.

v) Chsum5, which is a 2 byte checksum computed over the concatenation of (i), (ii), (iii) and (iv).

The concatenation of items (i) through (v) are encoded according to the selected scheme. The complete string contains either 23 or 28 bytes, depending on the encoding.

5) These three strings are transmitted to the RCST operator (e.g. read over the telephone or by file transfer).

6) On the RCST console, these three strings are entered. When base-32 encoding is used, the RCST should treat uppercase and lowercase as equivalent. If a string fails checksum, the operator is prompted to re-enter it.

7) Once the three strings are entered, the RCST operator is prompted for the passphrase. Operator enters passphrase. If the checksum fails, the operator is prompted to re-enter.

8) The AES key is determined and the ACC key is decrypted.

9) The RCST joins the network. At this point it receives the "next" ACC KEY in the normal way.

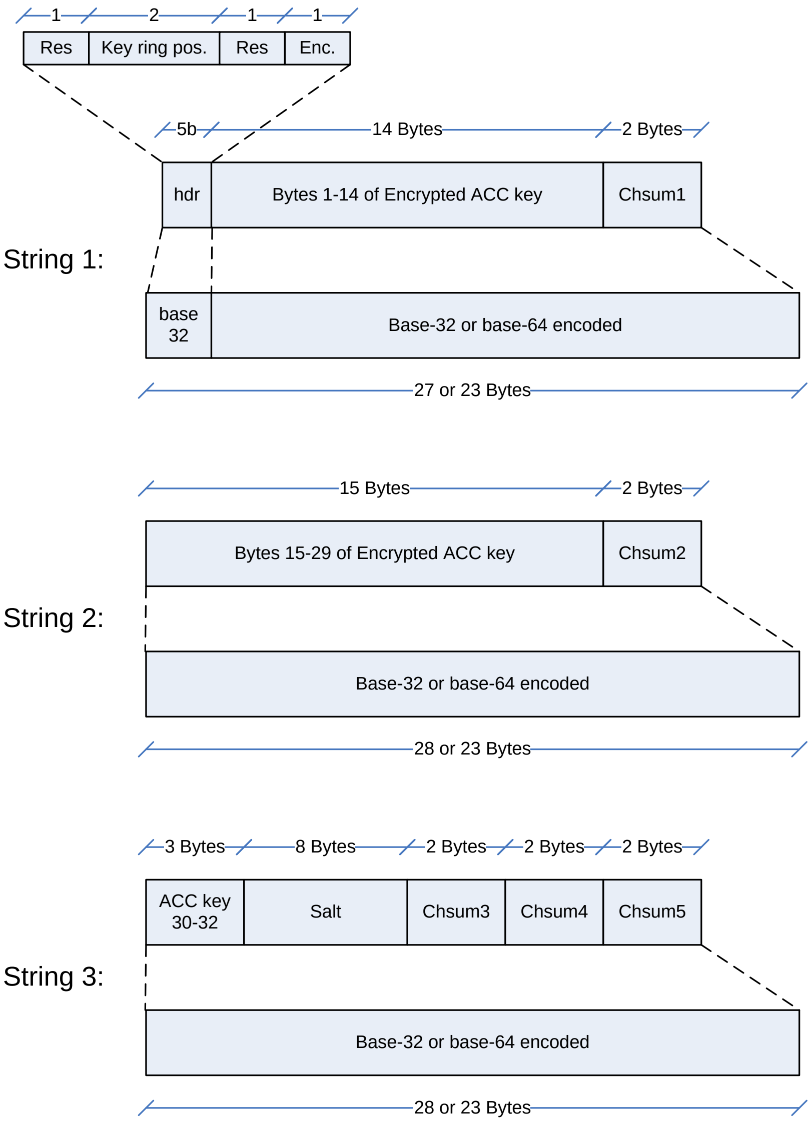
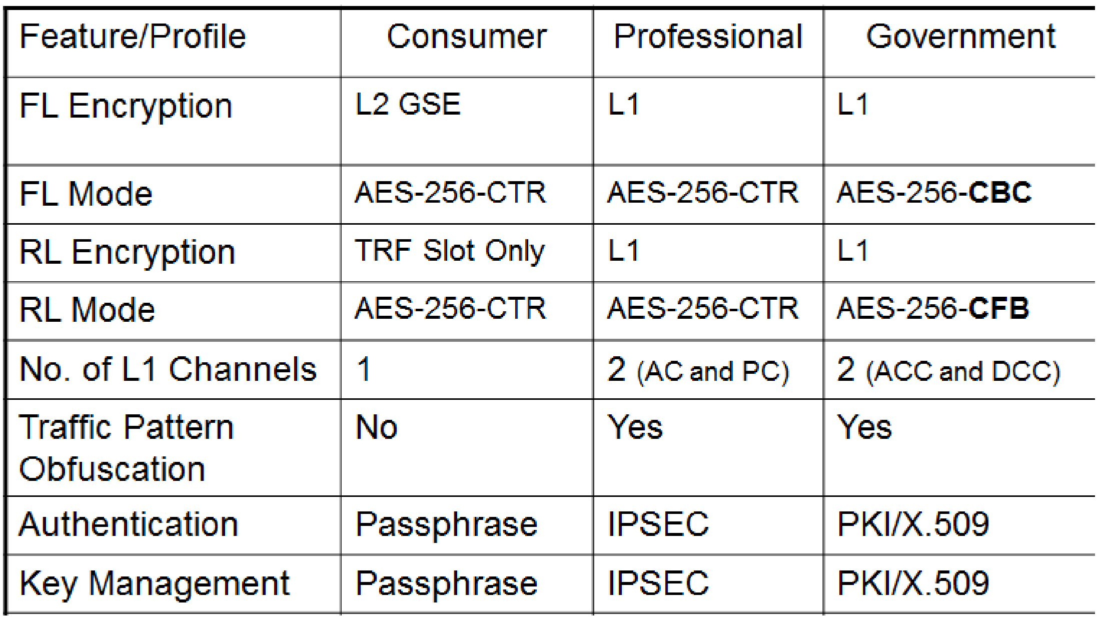


Figure 8.20: Formatting And Encoding Of Manual Key Entry Strings

8.5 Summary

8.5.1 Profiles Compared

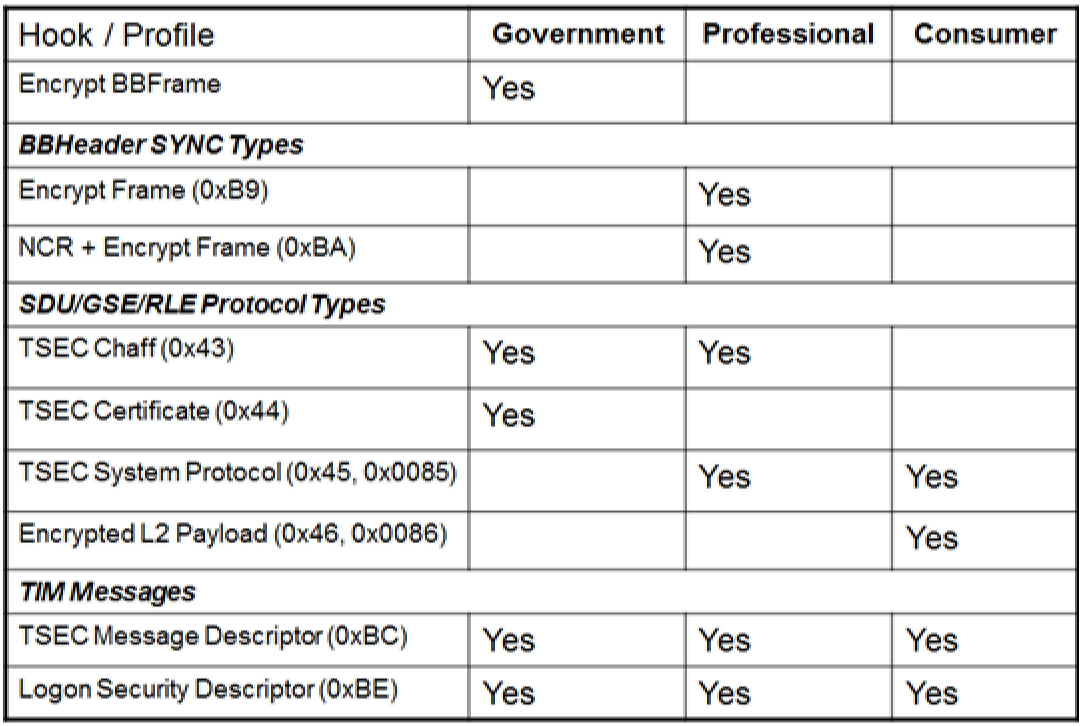
Table 8.24



Feature/Profile	Consumer	Professional	Government
FL Encryption	L2 GSE	L1	L1
FL Mode	AES-256-CTR	AES-256-CTR	AES-256-**CBC**
RL Encryption	TRF Slot Only	L1	L1
RL Mode	AES-256-CTR	AES-256-CTR	AES-256-**CFB**
No. of L1 Channels	1	2 (AC and PC)	2 (ACC and DCC)
Traffic Pattern Obfuscation	No	Yes	Yes
Authentication	Passphrase	IPSEC	PKI/X.509
Key Management	Passphrase	IPSEC	PKI/X.509

8.5.2 TRANSEC Hooks Used

Table 8.25: SUMMARY OF TRANSEC HOOKS USED



Hook / Profile	**Government**	**Professional**	**Consumer**
Encrypt BBFrame	Yes		
BBHeader SYNC Types			
Encrypt Frame (0xB9)		Yes	
NCR + Encrypt Frame (0xBA)		Yes	
SDU/GSE/RLE Protocol Types			
TSEC Chaff (0x43)	Yes	Yes	
TSEC Certificate (0x44)	Yes		
TSEC System Protocol (0x45, 0x0085)		Yes	Yes
Encrypted L2 Payload (0x46, 0x0086)			Yes
TIM Messages			
TSEC Message Descriptor (0xBC)	Yes	Yes	Yes
Logon Security Descriptor (0xBE)	Yes	Yes	Yes

9 RCST Deployment Guidelines

This clause provides guidelines material for the overall architecture and deployment of the RCST. Examples of frequency bands uses for transmit/receive, regulatory aspects as well as antenna alignment procedures and requirements are presented.

9.1 Example of RCST Architecture

The RCST typically complies with the architecture outlined in Figure 9.1. An RCST may conceptually consist of the Outdoor Unit (ODU), the Interfacility Link (IFL), and the Indoor Unit (IDU).

The ODU is composed of the following subsystems: Antenna Subsystem (ANT), Transceiver (TRx), and Mechanical Subsystem (MECH). The Interfacility Link (IFL) is a cable assembly, which interconnects the IDU with the ODU.

The ANT consists of the reflector(s) and a combined transmit/receive feed. Optionally the ANT may also include an additional receive feed for receiving from a satellite at a different orbital location. The receive (Rx) part of the TRx includes the Low Noise amplifier(s), frequency downconversion and polarization as well as frequency band selection. The transmit part (Tx) of the TRx performs frequency upconversion as well as power amplification. The MECH attaches the ODU to a firm structure and provides means for accurate pointing.

The IDU consists of the following subsystems: Network Interface Unit (NIU), User Interface Unit (UIU), Power Supply Unit (PSU) and Packaging. These subsystems can be implemented e.g. in a standalone IDU, within a desktop PC or Set Top Box.

The UIU is the interface between all receive/transmit elements of the IDU and the user device.

The NIU is constituted of at least one forward link receiver for reception of the forward link signalling (and the Traffic sent on the same carrier), a transmit chain for transmission of Traffic and forward link signalling to the ODU, and all the necessary controlling elements. If only one forward link receiver is available Traffic and forward link signalling should be received from the same carrier. Additional forward link receivers allow the transmission of Traffic and forward link signalling on different carrier. This results in significant improvement of operational flexibility and should be the preferred solution. The number of available forward link receivers is a parameter exchanged between the RCST and the NCC during RCST logon.

It should be noted that the conceptual split between IDU and ODU functionality as described above, and specified in the present document represents only one possible separation of functions. There may be also different approaches providing the same overall functionality.

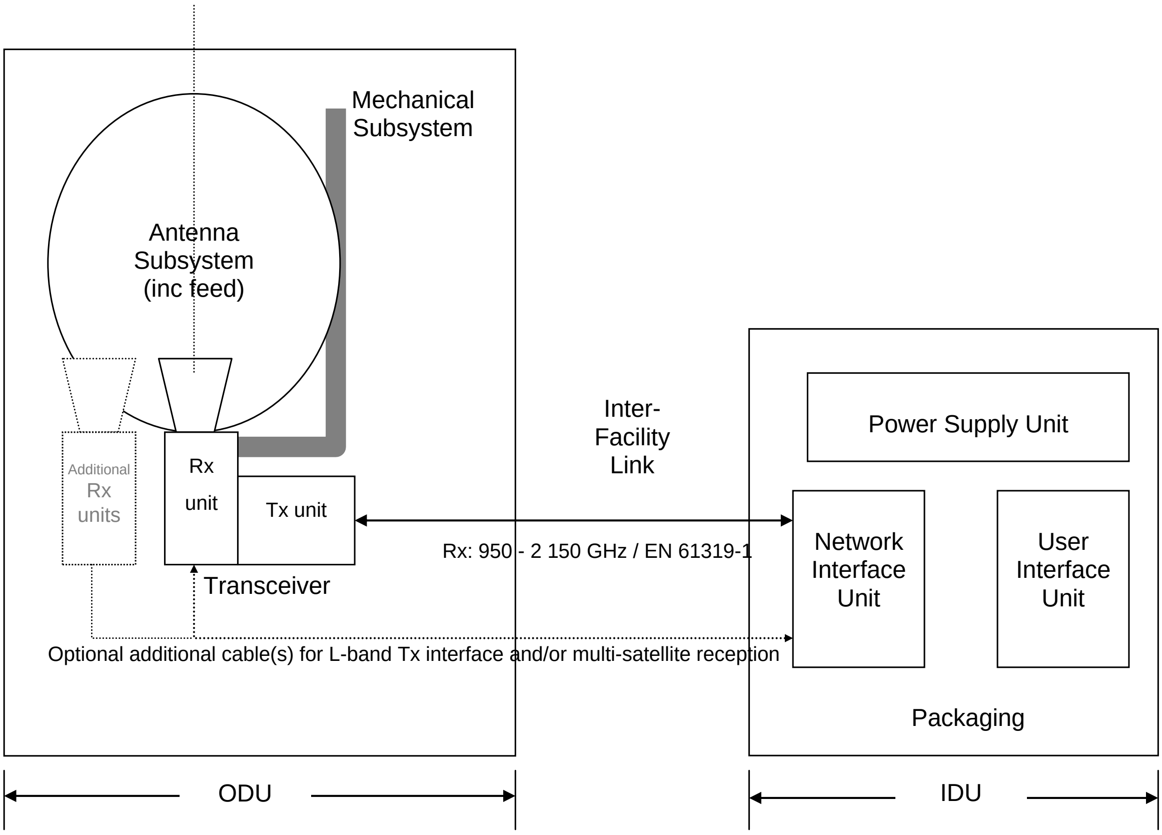


Figure 9.1: Conceptual RCST architecture

9.2 Frequency ranges and regulatory constraints envelope for Fixed RCST

9.2.1 Examples of used frequency bands for FSS

The following frequency bands are typically used:

- Reception is in one or several of the frequency bands of the Fixed Satellite Service (FSS) or Broadcast Satellite Service:

  10,70 GHz to 11,70 GHz.

  11,70 GHz to 12,50 GHz.

  12,50 GHz to 12,75 GHz.

  17,70 GHz to 19,70 GHz.

  19,70 GHz to 20,20 GHz.

  21,40 GHz to 22,00 GHz.

- Transmission is in one of the frequency bands allocated to FSS:

  14,00 GHz to 14,25 GHz.

  27,50 GHz to 29,50 GHz.

  29,50 GHz to 30,00 GHz.

Other bands are also envisaged. Regulation of usage of frequency bands is covered by other bodies.

Linear or circular polarization is used for transmission and reception.

9.2.2 Regulatory Aspects

There are four main sources of applicable regulatory documents:

- ITU-R, in the form of the Radio Regulations and specific recommendations.
- ETSI, which is not a regulatory body as such, but produces normative documents (EN's) that are adopted by reference in actual regulations.
- The FCC, which issues the regulations for the United States.
- CEPT, which issues common regulations for its (European) member states. These are sometimes, but not always, based on ETSI documents.

The regulations of most interest to the VSAT community deal with three subject areas:

- Interference limitation and co-existence.
- Electromagnetic compatibility.
- Blanket licensing.

The present document should not be used to justify the fulfilment of the essential requirements under article 3.2 of the R&TTE Directive [i.43]. Requirements for ElectroMagnetic Compatibility (EMC) under article 3.1b of the R&TTE Directive [i.43] are given in EN 301 489-12 [i.16]. Harmful interference is limited by requiring a minimum set of Control and Monitoring Functions (CMF) as well as specifying limits for on-axis radiation, off-axis spurious radiation, carrier suppression, off-axis EIRP emission density and pointing accuracy. These specifications are in general depending on the transmit frequency, regulatory authority and type of terminal. For example, for systems operating in Europe, EN 301 428 [i.7] applies in Ku-band and EN 301 459 [i.6] applies in Ka-band.

The RF parameters of the RCST have been selected to comply with the conditions identified in the ECC decisions ECC/DEC(06)02 [i.36] and ECC/DEC(06)03 [i.37] related to Exemption from Individual Licensing of High- and Low-EIRP Satellite Terminals. These ECC decisions make reference to the Harmonized Standards EN 301 459 [i.6] and EN 301 428 [i.7]. Furthermore, these ECC decisions add the following constraints:

- For low-EIRP terminals, EIRP not to exceed 34 dBW.
- For high-EIRP terminals:
  - EIRP less than or equal to a nationally defined limit, which can be in the range 50 - 60 dBW.
  - A coordination distance from airport perimeter fences that depends on the EIRP and station latitude; it varies between 500 and 3 900 m.

When operating at the nominal EIRP, the spectral regrowth should not exceed 20 dB. Spectral regrowth is defined as the ratio of the power in an adjacent channel of bandwidth (1+α) × symbol rate to the power in an equivalent bandwidth centred on the transmit carrier.

It should be noted that the frequency separation between the adjacent channel and transmit channel is system dependent.

9.3 Frequency ranges and regulatory constraints envelope for Mobile RCST

Operation of an RCST in a mobile environment is usually permissible under different regulatory and licensing conditions to those for the fixed or nomadic use. The regulatory conditions for mobile operation will, in general, impose constraints on the frequency bands employed, operational geographical area and off-axis emissions of the terminal. Furthermore, interference constraints may be imposed and require careful consideration.

This clause addresses the regulatory constraints for the use of mobile terminals, in particular terminals with small size antennas. It focuses on the earth-to-space direction, since on the space-to-earth direction, the necessary protection against FSS/FS interferences will be very dependent on the coordination situation and the adjacent systems characteristics, leading to specific constraints on the terminal sizing.

Within the ITU-R Radio Regulations, the following bands are allocated to the Mobile Satellite Service (MSS) in the earth-to-space direction and are thus of interest for DVB-RCS mobile applications:

5 925 - 6 425 MHz The Radio Regulations contain two footnotes (5.457A and 5.457B) concerning the use of ESV's and Resolution 902 (WRC-03) contains provisions relating to ESV's which operate in fixed-satellite service networks.

Notes:

5.457A: In the bands 5 925 - 6 425 MHz and 14 - 14,5 GHz, earth stations located on board vessels may communicate with space stations of the fixed-satellite service, in accordance with Resolution **902 (WRC-03)**.

5.457B: In the bands 5 925 - 6 425 MHz and 14 - 14,5 GHz, earth stations located on board vessels may operate with the characteristics and under the conditions contained in Resolution **902 (WRC-03)** in Algeria, Saudi Arabia, Bahrain, Comoros, Djibouti, Egypt, United Arab Emirates, the Libyan Arab Jamahiriya, Jordan, Kuwait, Morocco, Mauritania, Oman, Qatar, the Syrian Arab Republic, Sudan, Tunisia and Yemen, in the maritime mobile-satellite service on a secondary basis, in accordance with Resolution **902 (WRC-03)**.

14,0 - 14,5 GHz Secondary allocation in all three ITU-R Regions (earth-to-space).

29,5 - 29,9 GHz Primary allocation in Region 2 (earth-to-space).

Secondary allocation in Region 1 & 3 (earth-to-space).

29,9 - 31,0 GHz Primary allocation in all three Regions (earth-to-space), typically for government use only.

9.3.1 Regulatory constraints applicable to the Ku-band allocations

Within the Ku-band, only the sub-band 14,0 - 14,5 GHz is allocated to mobile satellite service on a secondary basis and covers the three types of utilization of the mobile services:

- land mobile satellite service (LMSS)
- aeronautical mobile satellite service (AMSS)
- maritime mobile satellite service (MMSS)

The transmissions from the Mobile Earth Station to the Satellite in the 14,00 GHz to 14,50 GHz band falling under a secondary allocation, the transmissions should not cause harmful interference to primary services (e.g. the Fixed Satellite Service (FSS)) and at the same time cannot claim protection from harmful interference from those services. In addition to FSS, some other terrestrial based services are using part of 14-14.5GHz Ku frequency band including the Fixed Services (FS) (in Regions 1 and 3), Radio Astronomy Services (RAS), and the Space Research Service (SRS) and these require appropriate protection from the mobile RCST emissions.

The use of this 14,0 - 14,5 GHz allocation was extended to the aeronautical mobile satellite service at the World Radiocommunications Conference in July 2003. This conference has also detailed the use of this band by ESV (Earth Station on board Vessels) through a new recommendation (Rec 37) and a new resolution (Res 902).

Within Europe, ETSI has developed several standards:

- EN 301 427 [i.8], harmonized EN for low data rate mobile satellite earth stations (MESs) except aeronautical mobile satellite earth stations, operating in the 11/12/14 GHz bands.
- EN 302 186 [i.9], harmonized EN for satellite mobile aircraft earth stations (AESs) operating in the 11/12/14 GHz bands.
- EN 302 340 [i.10], harmonized EN for satellite Earth Stations on board Vessels (ESVs) operating in the 11/12/14 GHz frequency bands allocated to the Fixed Satellite Service (FSS).

- EN 302 448 [i.11], harmonized EN for satellite Earth Stations on Trains (ESTs) operating in the 11/12/14 GHz frequency bands allocated to the Fixed Satellite Service (FSS).
- EN 302 977 [i.12], harmonized EN for Vehicle-Mounted Earth Stations (VMES) operating in the 12/14 GHz frequency bands.

These documents specify the minimum technical performance requirements of Mobile Station equipment with both transmit and receive capabilities for provision of mobile satellite service in the frequency bands given in Table 9.1.

Table 9.1: Frequency bands for the equipment specified in the standards

Mode of Operation	Frequency Band
Transmit	14,00 GHz to 14,50 GHz
Receive	10,70 GHz to 11,70 GHz
Receive	12,50 GHz to 12,75 GHz

Regulations regarding the USA: due to the adoption of 2 degree satellite spacing in the orbital arc over the USA the FCC has introduced regulations that are somewhat more stringent than the ETSI ones. The reader is referred to [i.13], Part 25 (Satellite Communications) of the USA Code of Federal Regulations (47: Telecommunications). In addition, the FCC 04-286 report and order provides some further background.

Within the ITU, Recommendation ITU-R M.1643 [i.14] is also of interest.

9.3.1.1 Off-axis EIRP limits

Considering the appropriate ETSI regulatory documents it can be seen that for directional antennas, the maximum EIRP in any 40 kHz band from any Mobile satellite Earth Station in any direction ϕ degrees from the antenna main beam axis should not exceed the following limits within 3° of the geostationary orbit:

$33 - 25 \log(\phi + \delta\phi) - 10 \log(K)$	dBW/40 kHz where	2,5°	$\leq \phi + \delta\phi$	$\leq 7{,}0°$;
$+12 - 10 \log(K)$	dBW/40 kHz where	7,0°	$< \phi + \delta\phi$	$\leq 9{,}2°$;
$36 - 25 \log(\phi + \delta\phi) - 10 \log(K)$	dBW/40 kHz where	9,2°	$< \phi + \delta\phi$	$\leq 48°$;
$-6 - 10 \log(K)$	dBW/40 kHz where	48°	$< \phi + \delta\phi$	$\leq 180°$;

where K is the number of simultaneous transmissions (K=1 for MF-TDMA system).

NOTE: These limits apply to satellites spaced at 3° apart. In the case of 2° spacing (reflected in Recommendation ITU-R S.728.1 [i.73]), a more constraining requirement - 8 dB less EIRP density- may be applied.

9.3.1.2 Particular constraints applicable to MMSS

The ESV terminal should have an antenna aperture greater than 1,2 meter (possibly 0,6 meter if agreed by the concerned licensing administrations).

9.3.1.3 Particular constraints applicable to AMSS

In region 1 (Europe) as well as Region 2 and 3, some countries operate Fixed Service (FS) links in the band 14,25 GHz to 14,50 GHz (shared band with FSS) on a primary basis. Since AES operation in the band 14,00 GHz to 14,50 GHz is on a secondary basis, there is a requirement for protection of Fixed Service (FS) systems in the band 14,25 to 14,50 GHz from in-band and out-band emissions from AES operating in the band 14,0 GHz to 14,5 GHz. The specification of protection of FS systems in the band 14,25 GHz to 14,50 GHz is based on the Power Flux Density (PFD) limits per AES. These limits are of a regulatory nature and only a small number of countries are employing FS systems in the band 14,25 GHz to 14,50 GHz. This requirement is applicable when the AES is in line of sight of a country employing FS systems, and could be relaxed if the operator of the AES network has an agreement with the Administration of that country.

When the AES should limit its PFD at the surface of the Earth, then in any 1 MHz bandwidth in the band 14,25 GHz to 14,5 GHz, the PFD at the surface of the Earth should not exceed the following limits:

$-132 + 0{,}5 \times \theta$ dB(W/m^2), where 0° ≤ θ ≤ 40°

-112 dB(W/m^2), where 40° < θ ≤ 90°

where θ (in degrees) is the angle of arrival at the Earth surface of the radio-frequency wave from the AES.

In addition, the AMSS being secondary to the Radio Astronomy service and to the SRS service (secondary in 14 - 14,3 GHz) according to Recommendation ITU-R M.1643 [i.14], protection of some specific Radio Astronomy stations in specific locations should also be considered.

Frequency management techniques using RAS/FS/SRS location knowledge may be used to perform active detection and mitigation of interferences.

9.3.1.4 Illustration of the impact of the off-axis EIRP constraint

It is believed that the main constraint for small size mobile terminals will come from the off-axis EIRP limit. This constraint is illustrated in Figures 9.2 and 9.3.

Assuming a theoretical Bessel shape antenna pattern, corresponding to uniform aperture illumination, it is possible to determine the maximum on-axis EIRP of the MES terminal as a function of the antenna diameter under the limitation of the off-axis EIRP described earlier.

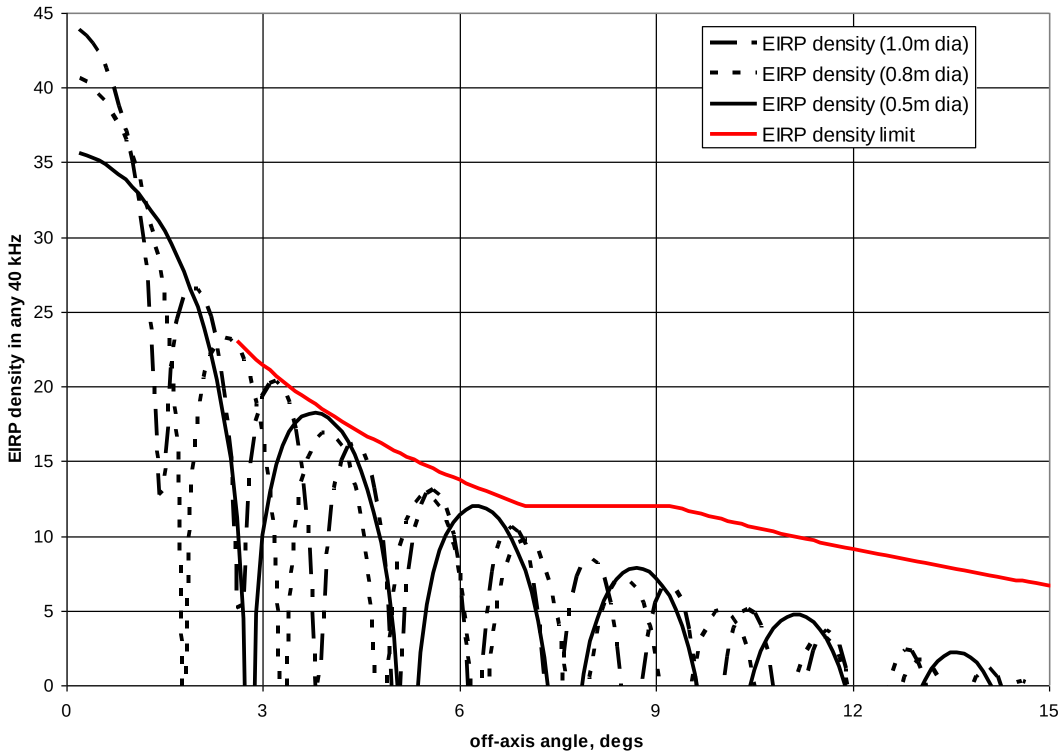


Figure 9.2: Off-axis EIRP density for large antennas

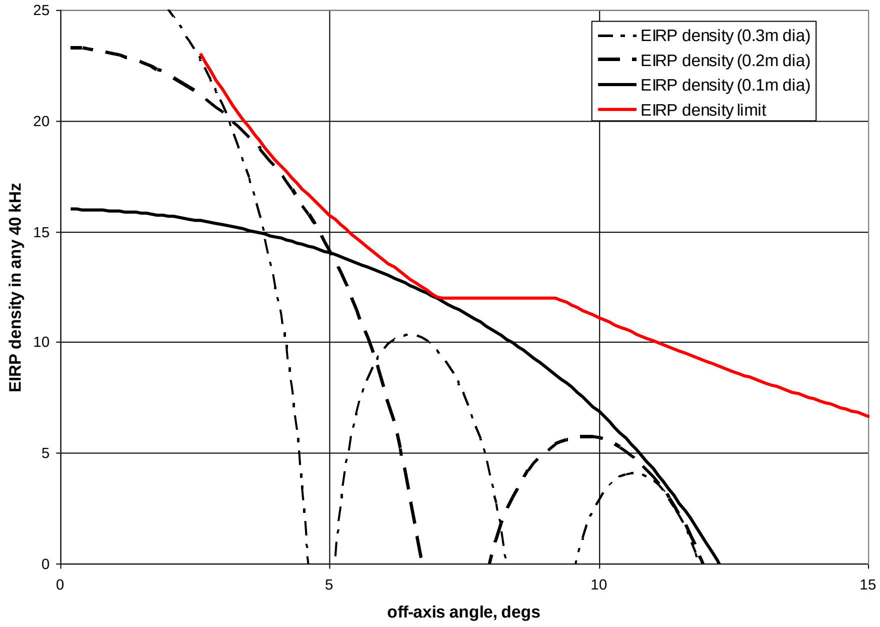


Figure 9.3: Off-axis EIRP density for small antennas

This EIRP off-axis mask is a significant constraint, since in order to close the link budget, the small size of the antenna cannot be compensated by an increase of RF power. Using tapered aperture illumination may, however, ameliorate the situation.

Figure 9.4 illustrates the evolution of the EIRP density as a function of the antenna diameter (assuming a theoretical antenna pattern as previously illustrated) under the constraints of not exceeding the EIRP off-axis mask limits.

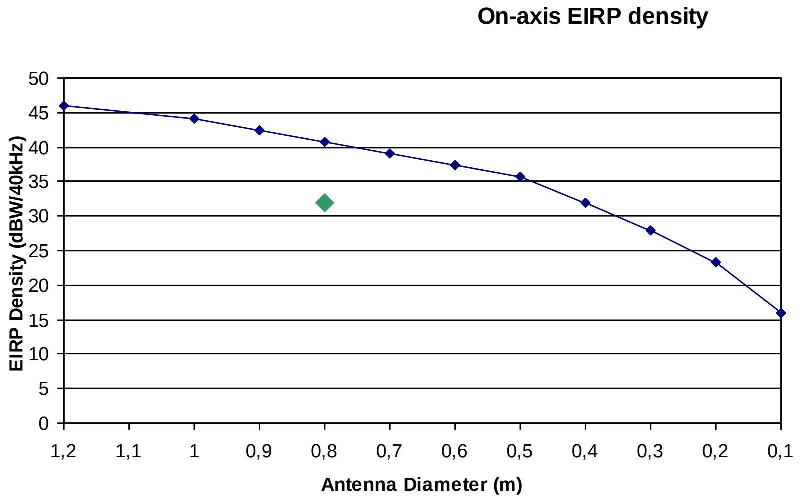


Figure 9.4: Evolution of the on-axis EIRP density as a function of the antenna diameter

As a reference, the EIRP density (in dBW/40 kHz) is provided in Figure 9.4 (i.e. 31,9 dBW/ 40 kHz) as well as the reference antenna size for this budget (80 cm). As a reference, the EIRP density (in dBW/40 kHz) extracted from reference link budgets is provided in Figure 9.4 (i.e. 31,9 dBW/ 40 kHz) as well as the reference antenna size for this budget (80 cm). It can be shown that in case small compact terminals (below 40 cm) are necessary, and for less favourable satellite coverage performances than the ones (resulting from higher terminal EIRP requirement), a reduction of the on-axis EIRP density may be necessary. In addition a small antenna size will require additional protection from receiving interference from adjacent satellite transmissions.

For those specific applications, the utilization of low code rate, or additional terminal return path and gateway forward path signal spreading may be considered. The latter option in particular is however clearly outside the provisions of the current standard.

9.3.2 Regulatory constraints applicable to the Ka-band allocations

No known regulations are applicable specifically to mobile applications in the Ka-band. The only known applicable standards are the following ETSI standards which relate to terminals in general:

- EN 301 358: [i.15]
- EN 301 459: [i.6]

A regulation for terminals on mobile platforms is in preparation; this will be published as EN 303 978 [i.74].

These standards state that the maximum EIRP in any 40 kHz band within the nominated bandwidth of the co-polarized component in any direction ϕ degrees from the antenna main beam axis should not exceed the following limits:

19 - 25 log ϕ - 10 log N	dBW	for	1,8°	≤	ϕ	≤ 7,0°;
-2 - 10 log N	dBW	for	7,0°	<	ϕ	≤ 9,2°;
22 - 25 log ϕ - 10 log N	dBW	for	9,2°	<	ϕ	≤ 48°;
-10 - 10 log N	dBW	for			ϕ	> 48°.

where N is the number of simultaneous transmissions (N=1 for MF-TDMA system).

The corresponding FCC regulation (FCC 25-138 [i.13]) is approximately 0,5 dB more stringent.

9.4 Interfaces

This clause describes examples of bidirectional communications used between the IDU and the ODU as Inter Facility Link (IFL).

For fixed satellite services the IFL usually takes the form of a pair of coaxial cables. Examples of implementation protocols using coaxial cables are described in clause 9.4.1 and annex G.

Particular VSAT systems may have the need for additional control protocols. For example, Mobile applications have the needs for control communications between the modem and the antenna, and will benefit from using TCP/IP over Ethernet. The control protocol may include features such as:

- Robust control interconnection of modem and antenna by using TCP
- Configuration of the antenna, particularly in the context of automatic beam switching
- Antenna position updates from the antenna (as an alternative for example to NMEA-0183)
- Communicating antenna status data and monitoring data (e.g. SNR, modem lock, TX power, blocking conditions)

Examples of such control protocols using Ethernet are outlined in clause 9.4.2.

Other ways of implementing the interface and control protocols than described here are also possible.

9.4.1 Coaxial Cable IFL

An example of the IFL protocol the Eutelsat DiSEqC™ bus specification [i.20].

In order to facilitate the use of RCST for individual or collective installation, the signals and the frequencies supporting those communications needs to be compliant with the EN 50083 [i.19] series and EN 61319-1 [i.18] if applicable.

Performance figures in this clause are not necessarily optimized in terms of the requirements of the normative document [i.1] for all potential implementations.

9.4.1.1 RX IFL

The RX IFL system interfaces between LNB and IDU. The definition is directly derived from the ETS 300 784 [i.17] and EN 61319-1 for "Universal" DBS/DTH terminals [i.18]. Table 9.2 shows the IFL parameters.

Table 9.2: IFL parameters

Parameter	Value	Unit	Note
Frequency scheme	no spectral inversion		
IF frequency input range, low band	See Table 9.3		
IF frequency input range, high band	See Table 9.3		
IF impedance	75	Ohm	
Return loss LNB & modem	> 8	dB	
Connector type	F-type		
Connector & cable color code	blue		
Cable loss @ 2 150 MHz	< 40	dB/100m	
LNB band switch tone command	according to EN 61319-1 [i.18]		
Low band selected	0,0 - 0,2	Vpp	18 - 26 kHz
High band selected	0,4 - 0,8	Vpp	
Polarization	fixed linear, orthogonal with TX (See note)		
DC supply voltage	11 - 19	V	on the LNB
DC supply current	< 300	mA	
NOTE: If dual-polarization reception is supported then the voltage (13/17V) switching command as specified in [i.18].			

Typical IDU IFL frequency input ranges includes the ones shown in Table 9.3.

Table 9.3: IFL Rx frequency

Frequency Band	RF [GHz]	LO [GHz]	IF [MHz]
C-Band	3,7 - 4,2	5,15	950 - 1 450
Ku-band - Low band	10,7 - 11,7	9,75	950 - 1 950
Ku-band - High band	11,7 -12,75	10,6	1 100 - 2 150
Ka-band	19,7 - 20,2		950 - 1 450

9.4.1.2 TX IFL

The TX IFL system interfaces between IDU and BUC. A single coaxial cable typically carries:

- The TX IF signal in L-band
- The TX LO frequency reference signal
- A low frequency sub-carrier for DiSEqC™ signalling
- The DC power supplying the BUC

In general L-band is recommended for all RCS terminals according to following scheme, with no spectral inversion, given in Table 9.4.

Table 9.4: IFL Tx frequency

Band	RF [GHz]	LO [GHz]	IF [MHz]
Ku-band	14,00 - 14,50	13,05	950 - 1450
Extended Ku-band	13,75 - 14,25	12,80	950 - 1450
Full extended Ku-band	13,75 - 14,50	12,80	950 - 1700
Ka-band	29,50 - 30,00	28,55	950 - 1450

It might be desired to ease installation by having the IDU automatically set the IDU output level to compensate for different cable attenuations and BUC gains. It is then recommended to use measured RF power from the BUC supported by the IFL protocol defined in annex G.

An alternative common method is to have a fixed cable loss IFL system and a fixed gain BUC. This method utilizes a standard level at the IDU output, a utilizing a well-known cable attenuation and well-known BUC gain. The IDU output level can also be calibrated for IFL cable loss and slope at the installation of the terminal by setting an applicable output level value in the IDU. Cable loss as well as the BUC gain are well known parameters that do not change over operational conditions or life-time of the terminal. Typical IFL cable characteristics are shown in Table 9.5.

Table 9.5: IFL cable characteristics

Parameter	Value	Unit	Note
IF drive level	set once during modem installation		
IF impedance	75	Ohm	
Return loss at BUC and IDU	> 13	dB	
Return loss cable	> 16	dB	
Connector type	F-type		
Connector & Cable color code	Red		
Cable attenuation @ 1700 MHz	< 30	dB/100m	
Cable attenuation uniformity	< 0,3	dB/MHz	
Cable length	< 50	m	

Recommended characteristics of the IDU LO reference are given in Table 9.6.

Table 9.6: IDU LO reference characteristics

Parameter	Value	Unit	Remarks
Reference type	frequency synchronous		
Frequency	10	MHz	sinusoidal
Frequency tolerance	< ±25	ppm	Overall (see note)
Level	0 ± 5	dBm	
Spurious level	< 30	dBc	0,01 - 20 MHz
Phase Noise @ 10 Hz	-86	dBc/Hz	
Phase Noise @ 100 Hz	-124	dBc/Hz	
Phase Noise @ 1 kHz	-134	dBc/Hz	
Phase Noise @ 10 kHz	-144	dBc/Hz	
Phase Noise @ 100 kHz	-152	dBc/Hz	
NOTE: The actual reference signal applied when the RCST has acquired NCR synchronization is derived from the NCR of the forward link, therefore it is highly accurate.			

9.4.1.3 ODU control signal

DiSEqC^{TM} IFL signalling between IDU and BUC enables a number of advanced applications and features. The signalling from IDU to ODU can be implemented by on/off voltage modulation, shown in Table 9.7.

Table 9.7: IFL signals

Parameter	Value	Unit	Note
Carrier frequency	22 ± 4	kHz	According to EN 61319-1 [i.18]
Modulation type	on/off using voltage superimposing		
Carrier level, logical 0/1	0 / 0,6	Vpp	

9.4.1.3.1 Concept of the 22 kHz Pulse Width Keying (PWK) Bus

The low data rate communication between the IDU and the ODU is based on a 22 kHz PWK signal as used by DiSEqC^{TM} [i.20]. The impedance of the bus at 22 kHz should be 15 Ω. A parallel inductor of 270 μH can be used to support a DC. power supply current. In this case a capacitor to ground should be supplied to shape the 22 kHz signal. The DC feeding point is grounded for 22 kHz with a capacitor. If a DC is not needed for powering peripheral devices, then in order to maintain correct operation of the DiSEqC^{TM} bus, there should be a minimum of 10 V bias applied, but the inductor and capacitor can be omitted.

The control signal from every device on the bus is produced by a 43 mA current shunt producing a 650 mV signal which is monitored by every device. This amplitude of the DiSEqC^{TM} carrier tone on the bus is normally too small to detect directly on a "TTL" or "CMOS" compatible pin on a microcontroller, so usually a "comparator" input, or a simple external (one transistor) amplifier, is required. In any case, it is important not to make the input too sensitive to small amplitude signals which may be "noise" or interference. It is recommended that the smallest amplitude normally detected is about 200 mV peak to peak. This can be achieved either with hysteresis (positive feedback applied around the comparator/amplifier) or with a DC bias offset (equivalent to about 100 mV) applied to the input of the amplifier/comparator. Hysteresis (if symmetrical) can maintain a reasonably constant 50 % duty cycle for the detected carrier tone, whilst the DC offset method may generate a less desirable asymmetric (pulse) waveform when the carrier amplitude approaches the lower limit.

All devices are connected in parallel on the bus and should therefore have a high impedance.

The PWK circuit specification is given in Table 9.8.

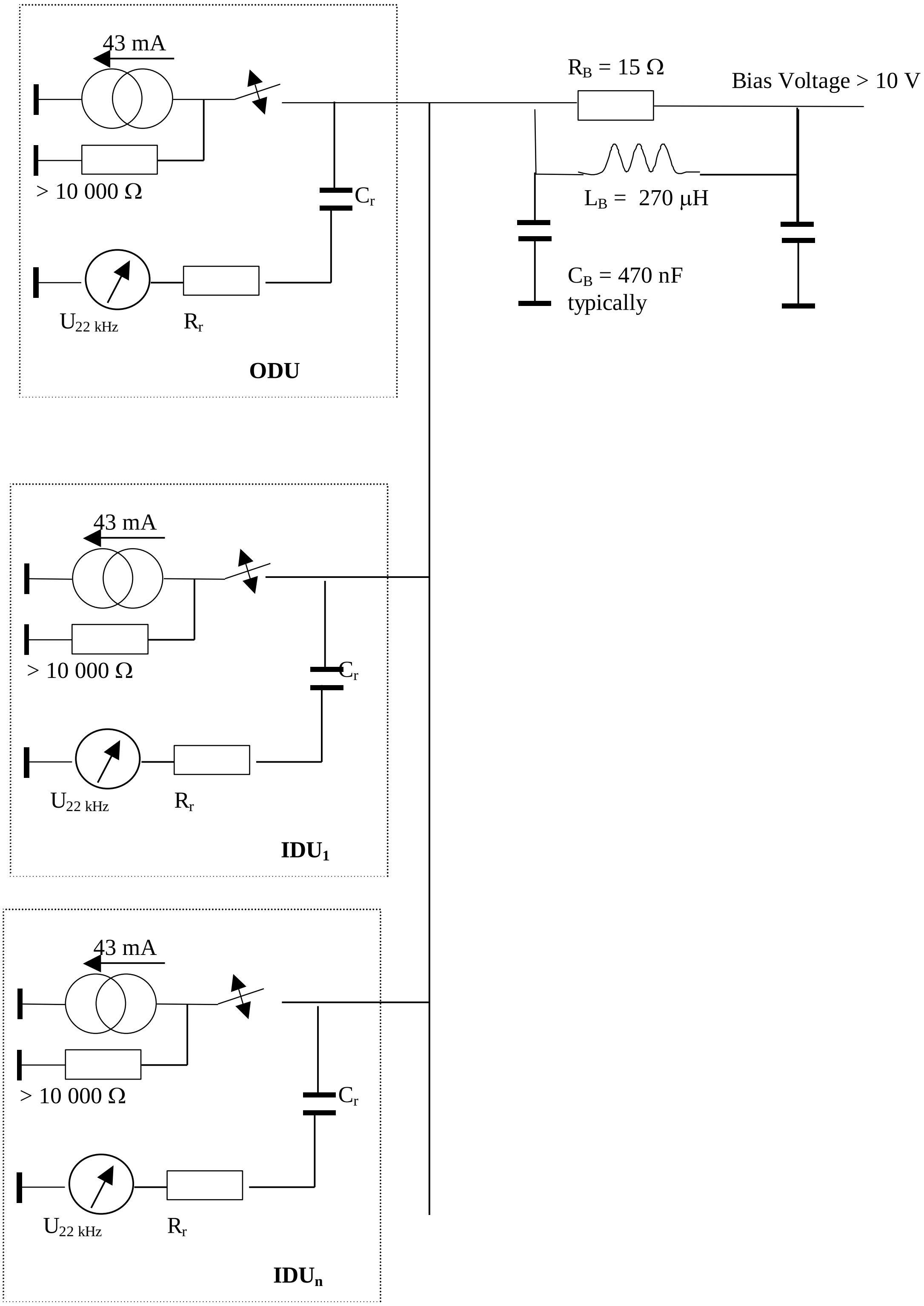


Figure 9.5: 22 kHz PWK bus concept

Table 9.8: PWK circuit specification

Parameter	Value	Unit	Note
Carrier frequency	22	kHz	±20 %
Bus load impedance R_B	15	Ω	±5 %
DC supply			
Bus load inductance L_B	270	µH	±5 %
Bus load capacitance C_B	470	nF	typical
Current source			
current amplitude	43	mA	±10 %
source impedance	> 10	kΩ	
22 kHz carrier detection device resistance R_r	5 to 10	kΩ	typical
DC block capacitor	Typically a few nF, but depends on the value R_r, it should be chosen so as to give a time constant of around 100 µS		
Bit definition			
timing base	0,5	ms	±0,1
bit length	1,5	ms	
"0"	1,0 ms burst + 0,5 ms pause		
"1"	0,5 ms burst + 1,0 ms pause		

9.4.1.4 Control functions from the IDU

The following functions should be available:

- SSPA ON/OFF (relates to disabled state as described in EN 301 459 [i.6].
- Tx Unit power off.

The following functions may be available:

- Frequency tuning within the wide frequency range of the slow frequency agility (if applicable).
- Software update of the ODU.
- Tx Frequency band selection (select different Local Oscillators).
- Modulation ON/OFF (transmit Continuous Wave).
- Set Transmit output power level.
- Get ODU location data (latitude and longitude).
- Full Reset.
- Software Reset.
- Password Reset.

The transmit control signal level at the output of the IDU should comply with the clause 5.3.2 of EN 50083-10 [i.19].

9.4.1.5 Monitoring functions (from ODU on request)

The following functions should be available:

- SSPA ON/OFF status.
- Phase lock oscillator status.
- Power supply status.
- Device status.
- ODU manufacturing information.

The following functions may be available:

- ODU temperature information for compensation.
- ODU output power level for compensation.
- ODU RF calibration parameters for RF level detector compensation.
- LNB status (in the case the LNB or a part of the LNB is controlled by the ODU control bus).
- Get ODU location data (latitude and longitude).
- Authentication information exchange between the ODU and the IDU.

9.4.1.6 Control and Monitoring protocol description

An example of the protocol description is provided in annex G.

9.4.2 Control Interface using Ethernet

Many mobile applications have needs for communication between the modem and antenna controller that are more sophisticated than what is commonly used for FSS. Such communications can serve a number of purposes. One important element is facilitation of compliance with regulatory requirements, such as reporting of mispointing and fault conditions and the transfer of logged parameter values. Another important aspect of this communication is the ability to instruct the modem to cease transmission when the antenna controller has detected a situation where this is required - it is typically the antenna controller that will have position information etc. for making these types of decisions.

Other elements of this communication can facilitate installation and optimal operation, such as informing the modem about transmitted power, antenna pattern and polarization skew.

It is not always practical to carry this communications over the coaxial IFL cables. There alternative industry-standard interfaces available, which typically use dedicated serial lines or Ethernet connections.

One such protocol is the VSAT Antenna Control Protocol (VACP^{TM}) that is supported by several antenna vendors. The VACP^{TM} specification can be acquired through (http://stmi.com/pubdoc/vacp).

Another protocol is OpenAMIP^{TM}, also adopted by several manufacturers of mobile antenna units. OpenAMIP^{TM} is freely available through the Internet (http://www.idirect.net/Products/iDirect-Intelligent-Platform/Mobility.aspx). It has also been incorporated in the ARINC-791 standard for aeronautical terminals (see www.arinc.com).

9.5 ODU environmental conditions

9.5.1 Operational environment

There should be no significant degradation of the specified system and subsystem performance when operating under the following environmental conditions:

Temperature:	-30 °C to +50 °C.
Solar Radiation:	500 W/m^2 max.
Humidity:	0 % to 100 % (condensing).
Rain:	up to 40 mm/h.
Wind:	up to 45 km/h.

The ODU mechanical construction should make sure there are no random vibrations during wind conditions which cause any significant performance degradation.

9.5.2 Survival conditions

The ODU is allowed to degrade in performance, but should maintain pointing accuracy and suffer no permanent degradation under the following environmental conditions:

Temperature:	-40 °C to +60 °C.
Solar Radiation:	1 000 W/m^2.
Humidity:	0 % to 100 % (condensing).
Precipitation:	up to 100 mm/h of rain or 12 mm/h of freezing rain or 50 mm/h of snowfall.
Static load:	25 mm of ice on all surfaces.
Wind:	up to 120 km/h.

Storage and Transportation

Temperature:	-40 °C to +70 °C.

Shock and Vibration: as required for handling by commercial freight carriers.

9.6 RCST Antenna Alignment Guidelines

The normative document [i.1] provides some signalling support that may be used to establish the necessary information exchange between the RCST and the NCC to facilitate manual and automated alignment of the RCST antenna as summarized below.

The antenna alignment is composed of forward link alignment and return link alignment processes. The return link alignment starts only after the forward link alignment is completed with a level of accuracy that is specified by the NCC. The subject level of accuracy is periodically broadcast by the NCC as minimum forward link SNR value before the return link transmitter can be activated.

The return link alignment process uses probing signals on the return link. These probes may be in the form of Continuous Wave (CW) transmissions or Installation Burst (IB) transmissions. The NCC periodically broadcasts the types of return link alignment methods that it supports. The RCST indicates in the initial Logon SDU the types of return link alignment methods that it can perform. The NCC selects and indicates in the Logon Response the alignment method that the RCST should implement during the return alignment process. In addition, the NCC may indicate to the RCST an alignment population ID that is different from the RCST's operational population ID. The alignment population ID is used by the RCST together with the RMT signalling table so as to identify the Forward Link Signalling (FLS) the RCST should tune to during the alignment process.

The NCC sends unicast feedback and commands to the RCST in response to return link alignment probes. The feedback may include CNR measurement, cross-polar and co-polar measurements, cross-polar and co-polar thresholds, and alignment status {Failure, Success, In-progress}. The commands may include configuration information in regards to the return link alignment probes.

- For CW alignment, configuration includes transmission frequency, start time, duration of transmission, and transmission EIRP. Note that the NCC may exclude EIRP configuration if dynamic EIRP is not used. If the RCST is a fixed-EIRP device and if the NCC dictates dynamic updates on the EIRP, the RCST should terminate the alignment procedure with failure.

- For IB alignment, configuration includes the bit pattern that the RCST should use in the Installation Burst. Note that CW transmission configuration also conveys resource allocation information in the form of {frequency, start time, duration} configuration. In contrast, in the case of IB alignment, configuration and resource allocation take place in different signalling tables. The installation bursts are to be transmitted in dedicated access logon slots, and the NCC allocates such slots to the RCST in the burst time plan (SCT/FCT2/BCT/TBTP2 tables).

- For both IB alignment and CW alignment, the NCC may set a remaining duration timer in RCST at the expiry of which the RCST should terminate the return link alignment process. This is a safety precaution to preclude the possibility that the RCST ends up in a state where it transmits on the return link indefinitely. If necessary, the NCC may send a new remaining duration value to restart this timer in the RCST.

The NCC is in complete control of the return alignment process. Hence, the NCC may deny, terminate, re-start, and modify the return alignment process if necessary. The NCC may also broadcast contact information in ASCII format of a Network Operations Centre for assistance during the alignment process.

9.6.1 Motorised antenna control for automated alignment

It is possible to develop intelligent search algorithms to calculate the elevation and azimuth of the RCST antenna as a function of feedback received from the NCC - in both CW and IB-based return link alignment. Such search algorithms may minimize installation duration and may remove the need for manual installation when combined with motorised antenna control equipment.

It should be noted that there is a master-slave relation between the NCC and the RCST during return link alignment. RCST-side protocols that may drive the return link alignment procedure should only transmit on the return link when Pointing Alignment Support descriptors from the NCC command so. There may be timers included in these protocols, at the expiry of which the RCST goes into the initial state (power-down), wait for a random period, and retry the whole alignment procedure from power-up.

9.6.2 RCST Alignment Accuracy

9.6.2.1 Required Forward Link alignment accuracy

Before starting the return link alignment procedure it is necessary to achieve the required minimum pointing accuracy through the forward link alignment procedure.

The term "alignment" here includes two different aspects:

- alignment of the antenna bore sight towards the satellite (both circular and linear polarizations);
- alignment of the feed to the correct tilt angle (linear polarization only).

Two constraints apply to the forward link pointing accuracy requirement:

- it has to ensure that the installation burst transmission does not cause harmful interference to other systems;
- it has to allow the RCST correctly performing the MAC logon, in the worst ST condition (at the EoC).

The radiation pattern of the User Terminal antenna is modelled analytically as a paraboloidal reflector. The resulting radiation pattern is given by:

$$f_y = \pi a^2 E_0 \cdot \left[2(1-A)\cdot \frac{J_1(u)}{u} + A \cdot 2^n \cdot n! \cdot \frac{J_{n+1}(u)}{u^{n+1}} \right],$$

where:

- *a* is the antenna radius,
- *A* is the Tapering Amplitude; A = 0,6838 in this analysis (corresponding to an Edge Taper of 10 dB),
- *n* is the roll off depending on the feed radiation pattern; n=1 is considered in this analysis,
- J_j is the Bessel function of the first kind and order *j*.

Concerning the cross-polar off-axis EIRP the analysis aims to identify a requirement on the feed rotation accuracy achieved after the forward link alignment procedure.

The constraints associated to such a requirement are two:

- the polarization mismatch loss, affecting the link budget during the logon and return link alignment procedures;
- the regulatory constraints on the cross polar EIRP emission.

To assess the impact of the feed rotation error, it has been assumed that the terminal antenna is perfect, therefore all the contribution to the cross polar component depends just on the rotation error. Under this assumption, the power on the cross-polar component can be related to the rotation error τ through the following equation:

$$G_{cross} = G \cdot \left(1 - \cos^2(\tau)\right)$$

Where G is the antenna pattern under perfect rotation conditions.

On the other end, the loss due to the polarization mismatch on the cross-polar component is evaluated as:

$$Loss_{polarization} = \cos^2(\tau)$$

Apart from the regulatory constraints, it is worth mentioning that in case the network itself is planned to exploit the two orthogonal linear polarizations, then the feed rotation accuracy should be specified according to the required Cross Polarization Discrimination (XPD) performance.

A typical XPD value for a commercial antenna with perfect alignment is in 20 - 25 dB range.

Using a simple analytical model it can be derived that the XPD due to the feed rotation accuracy is proportional to the squared tangent of the rotation error.

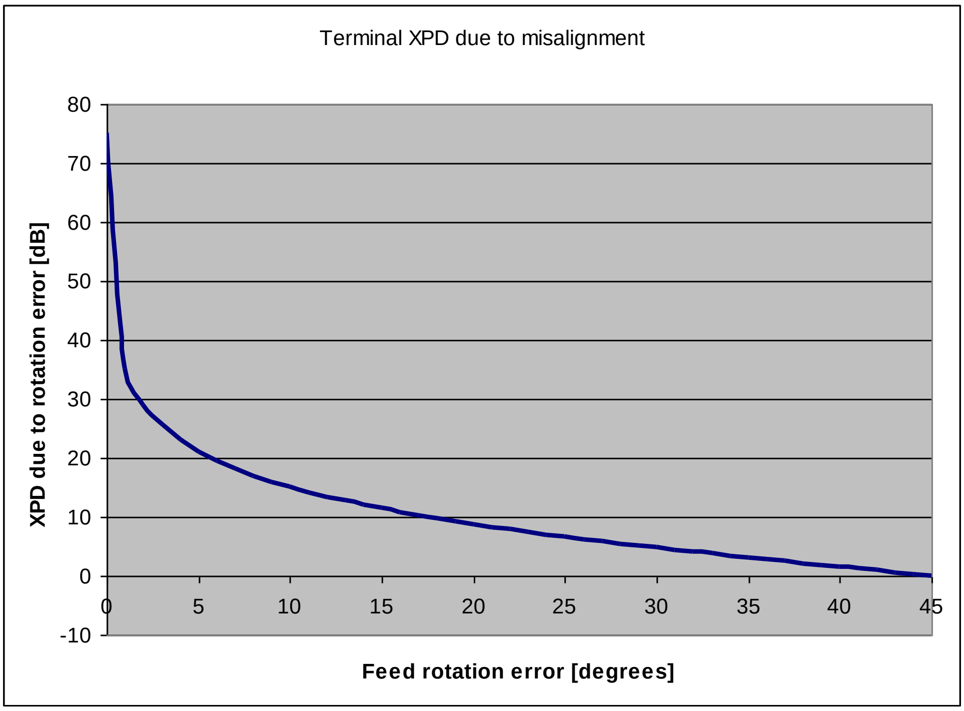


Figure 9.6: XPD due to feed rotation error

Therefore assuming to tolerate a degradation of 0,5 dB in the XPD due to the rotation error (assuming an XPD of 25 dB in perfect alignment conditions), the XPD due to the rotation error should be kept higher than 34 dB, corresponding to a feed rotation error lower than 1,1 degrees.

9.6.2.1.1 Ku Band System Scenario

In the Ku-band uplink scenario two terminal types are considered: a consumer terminal, with a 0,96 m antenna dish size and a BUC with 2 W maximum output power, and a professional terminal with a 1,2 m dish and a BUC with 3W maximum output power. Figure 9.7 shows the radiation pattern of the two antennas at 14 GHz.

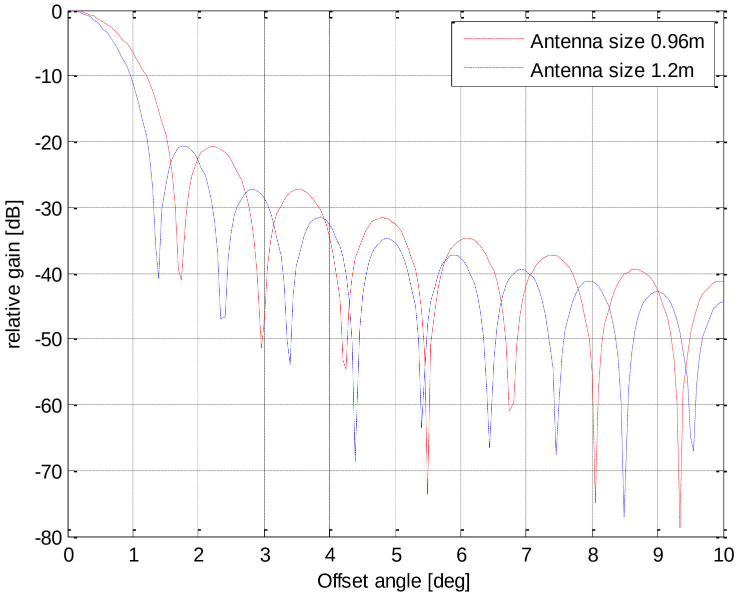


Figure 9.7: Analytical Radiation Pattern vs ϕ – Ku band terminals

The symbol rate of the installation burst is set to 64 ksymbols/s, and the G/T at the EoC is -4 dB/K. The feed rotation accuracy is assumed to be better than 15 degrees.

Starting from this system scenario, the operating point of the power amplifier, i.e. the Output Back Off (OBO) is derived ensuring to meet the Off-axis EIRP emission density limit within the band.

Figure 9.8 shows the off-axis EIRP for the two terminal types described above with perfect alignment.

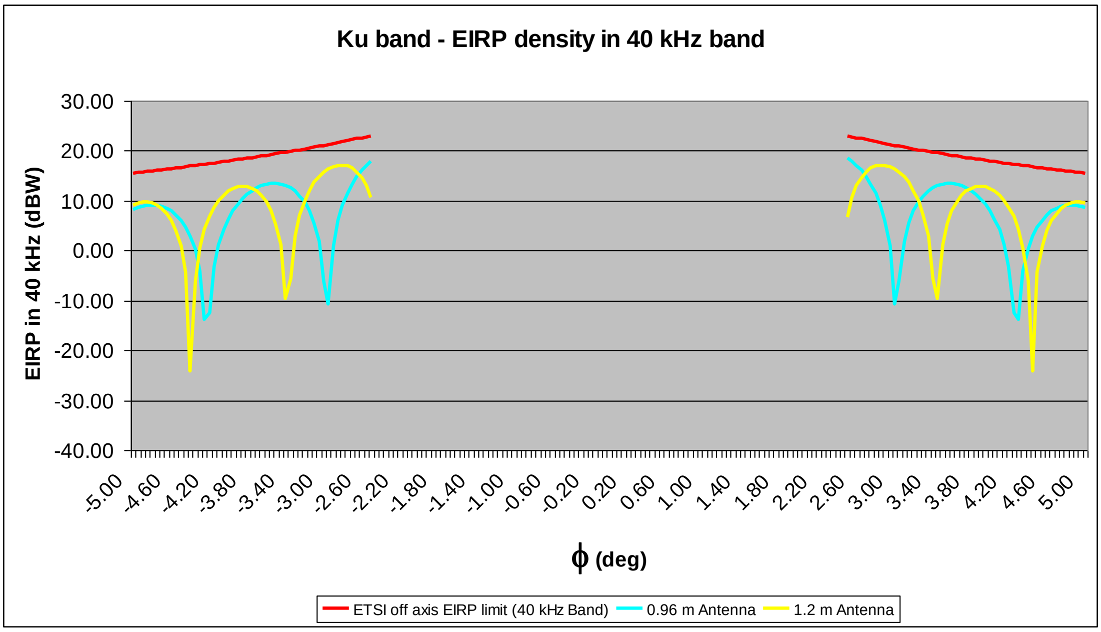


Figure 9.8: Off-axis EIRP, Ku band terminals - perfect alignment

Figure 9.9 shows the OBO settings (the back-off with respect to the nominal operating point, that is the saturated power minus a nominal Back Off of 0,5 dB).

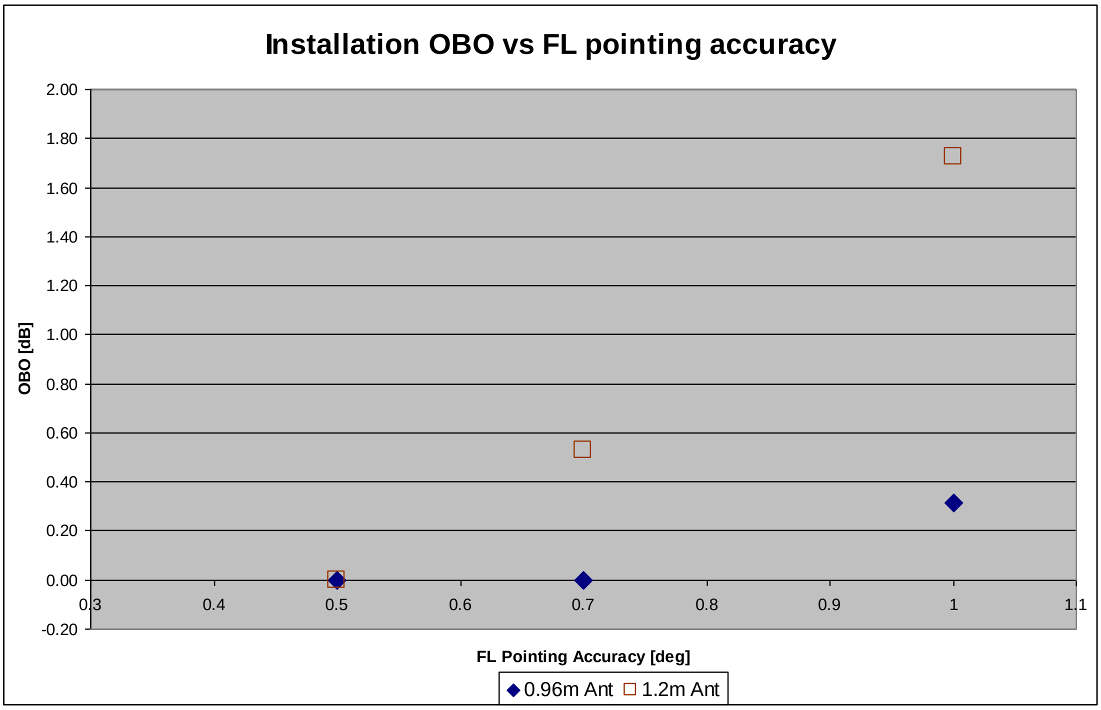


Figure 9.9: OBO settings - Ku band system scenario

Figures 9.10 and 9.11 show the resulting co-polar EIRP (in the worst 40 kHz band) for the two terminals, for a forward link alignment accuracy of 1, 0,7 and 0,5 degrees.

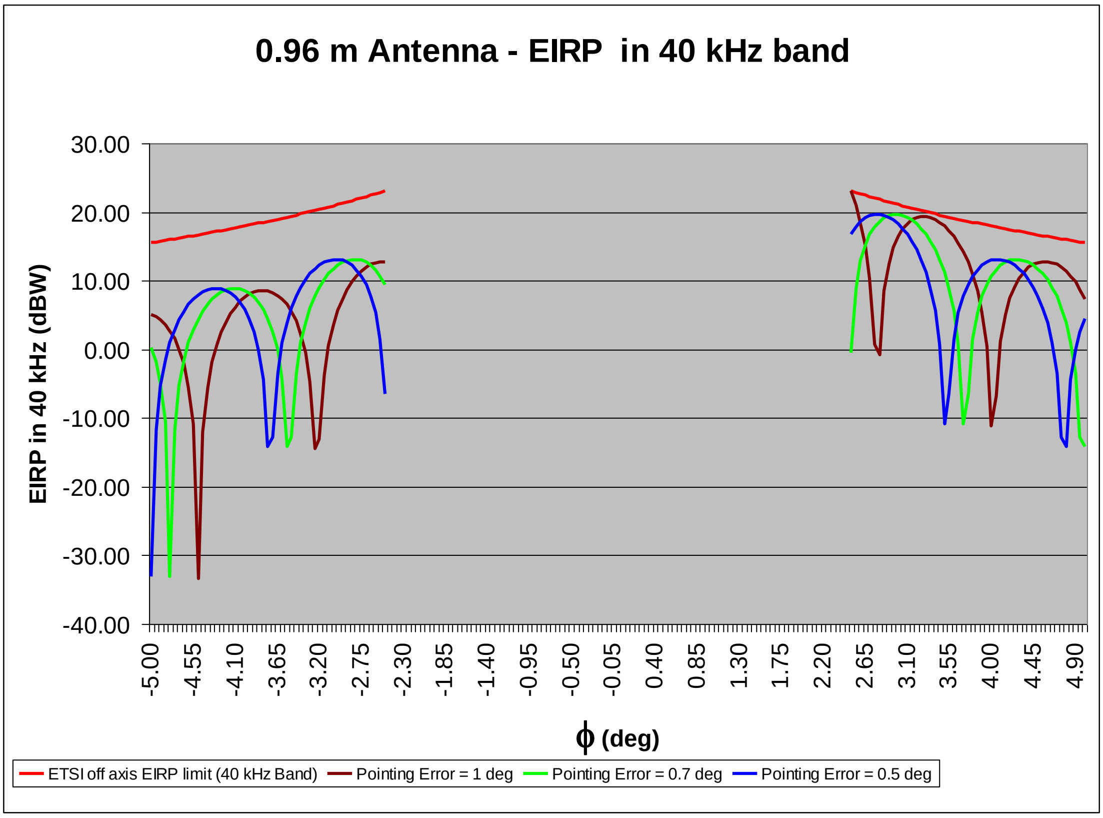


Figure 9.10: Off-Axis EIRP in 40 kHz, 0,96 m Ku band antenna

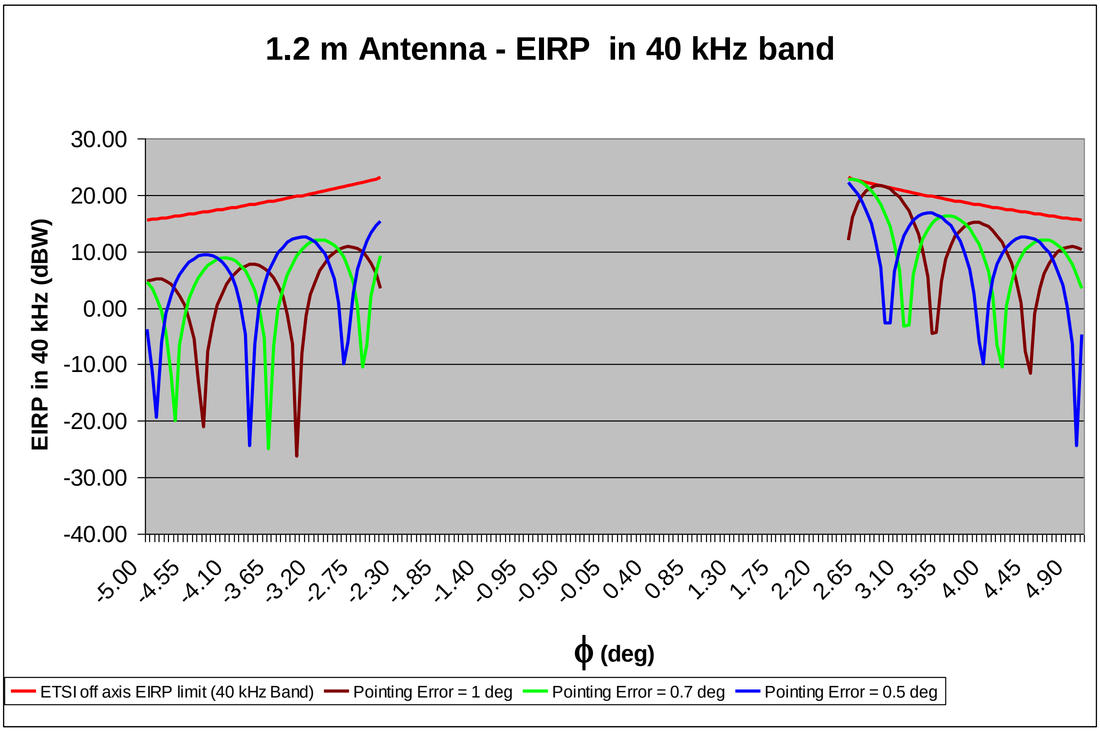


Figure 9.11: Off-Axis EIRP in 40 kHz, 1,2 m Ku band antenna

Applying these OBO settings derived from the co-polar EIRP limit, the cross-polar EIRP is estimated as shown in Figures 9.12 and 9.13 for the 0,96 and 1,2 m antenna terminals respectively.

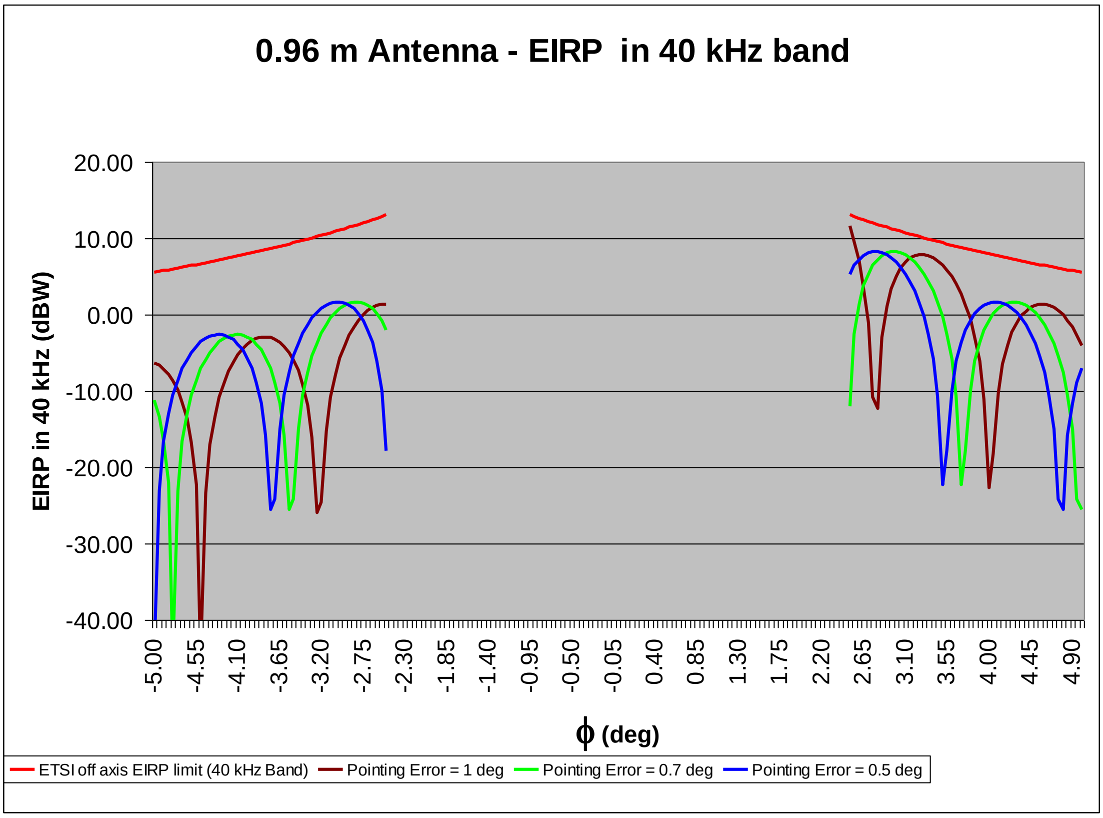


Figure 9.12: Cross-polar Off-Axis EIRP in 40 kHz, 0,96 m Ku band antenna

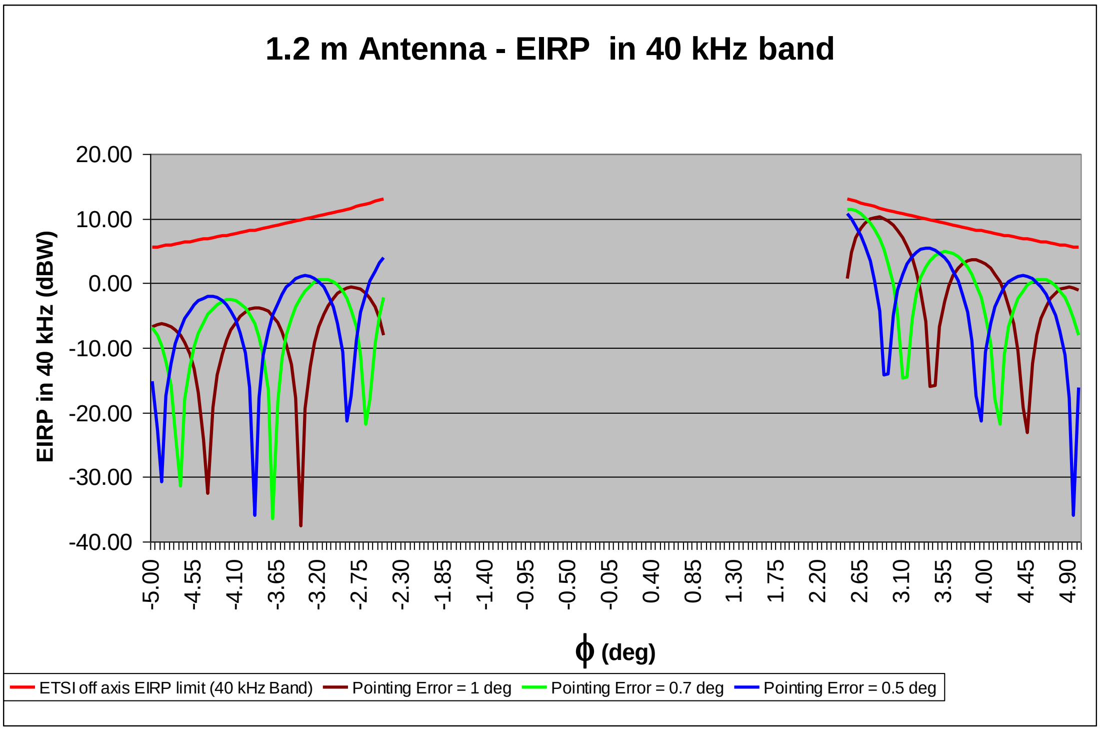


Figure 9.13: Cross-polar Off-Axis EIRP in 40 kHz, 1,2 m Ku band antenna

Therefore also the cross-polar mask is respected, as the co-polar limitation turns out to be the most stringent one. As shown in Figure 9.6, 15 degrees feed rotation error corresponds to an XPD of around 11 dB. Since the off-axis EIRP mask for the cross polar component is 10 dB lower than that of the co-polar component, the cross polar mask is automatically fulfilled when the co-polar is respected.

Figure 9.14 shows the link budget results in terms of Link Margin. The installation burst is transmitted with QPSK modulation, and FEC rate 1/3 turbo code. For both terminals, it is possible to successfully perform the MAC logon and meet the off-axis EIRP limits with a pointing error up to 1 degree after the forward link alignment procedure is completed.

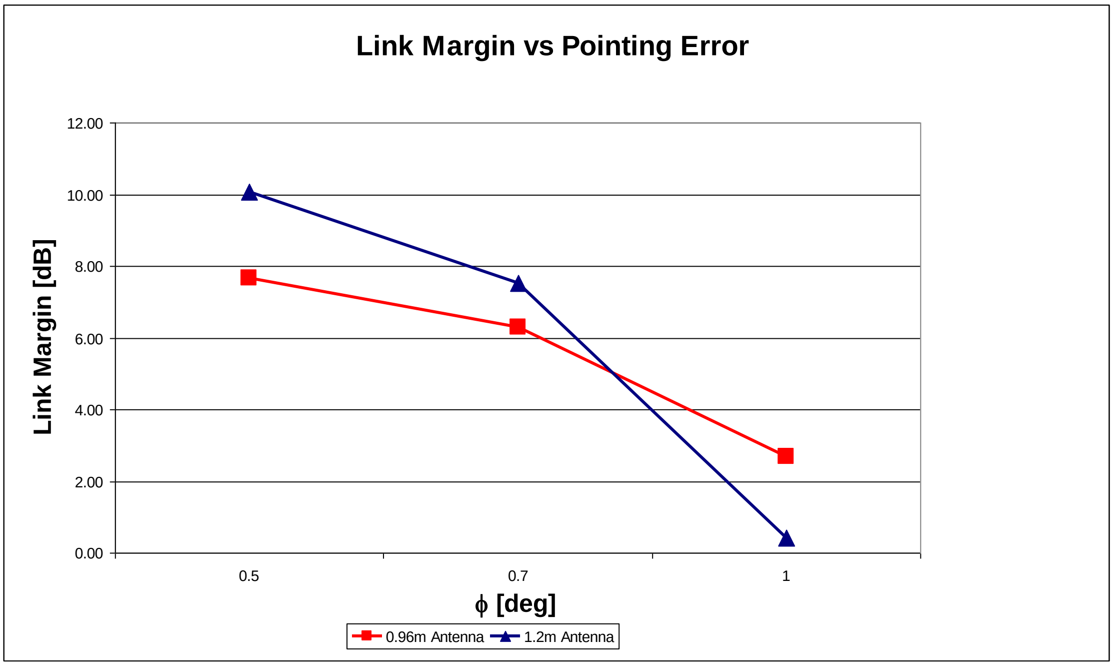


Figure 9.14: Link Margin versus forward link alignment accuracy

9.6.2.1.2 Ka Band System Scenario

Three different antenna diameters are considered: 0,45, 0,6 and 1,2 meters. Figure 9.15 shows the resulting radiation patterns.

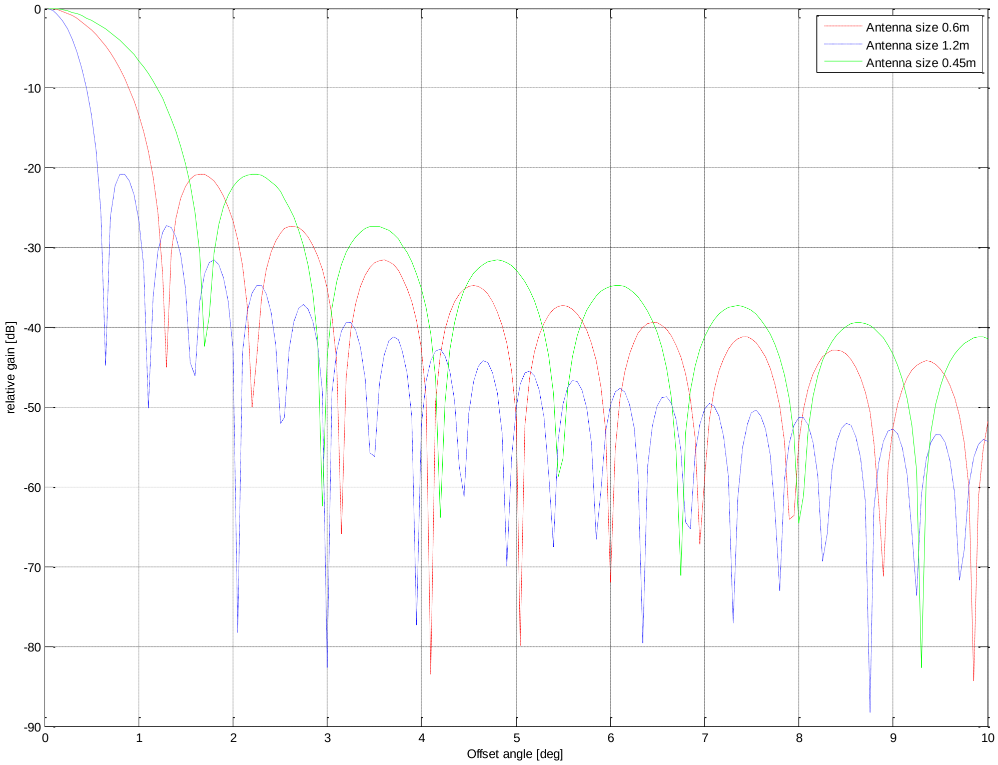


Figure 9.15: Analytical Radiation Pattern vs ϕ – Ka band terminals

The symbol rate of the installation burst is set to 160 ksymbols/s, and the G/T at the EoC is 15,2 dB/K.

As in the previous analysis, the off-axis EIRP for the three considered terminals in case of perfect alignment is used to derive the OBO settings.

Note that if a terminal exceeds the EIRP mask, than it cannot boost all the power on 160 Ksymbols/s carriers.

As for the Ku band scenario, the feed rotation accuracy is assumed to be better than 15 degrees.

With the derived OBO settings, the resulting EIRP (in the worst 40 kHz band) for the three antenna sizes, meets condition 1. The same conclusion applies for the cross-polar pattern, as for the Ku band scenario.

Figure 9.16 shows the link budget results in terms of Link Margin. The logon burst is transmitted with QPSK modulation, and FEC rate 1/3 turbo code. These results are independent of the carrier symbol rate, as long as the OBO setting is greater than zero.

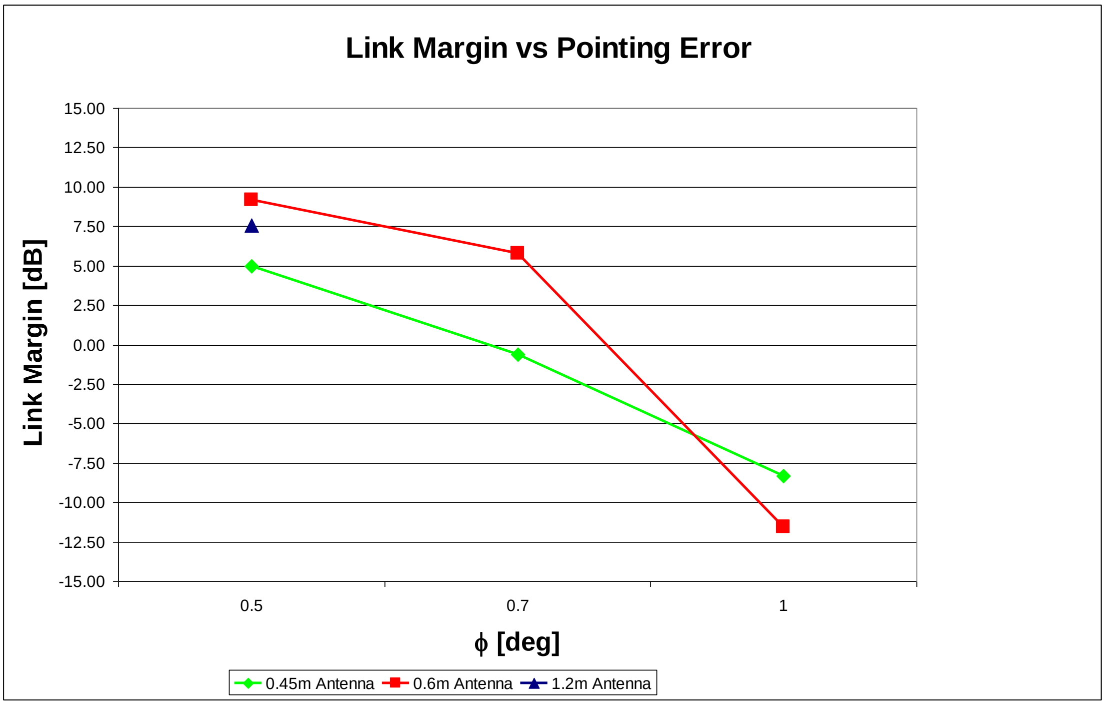


Figure 9.16: Link Margin versus forward link alignment accuracy

It can be seen that for forward link alignment accuracies better than 0,5 degrees, the link budget can be closed with a margin larger than 5 dB for all of the three antenna types.

For the 1,2 meter reflector, the lower accuracies are not considered as they would exceed the 3 dB beam width θ_{3dB} (as a rule of thumb it also correspond to the angle between the boresight and the first null). This could cause a wrong alignment to the closest secondary lobe.

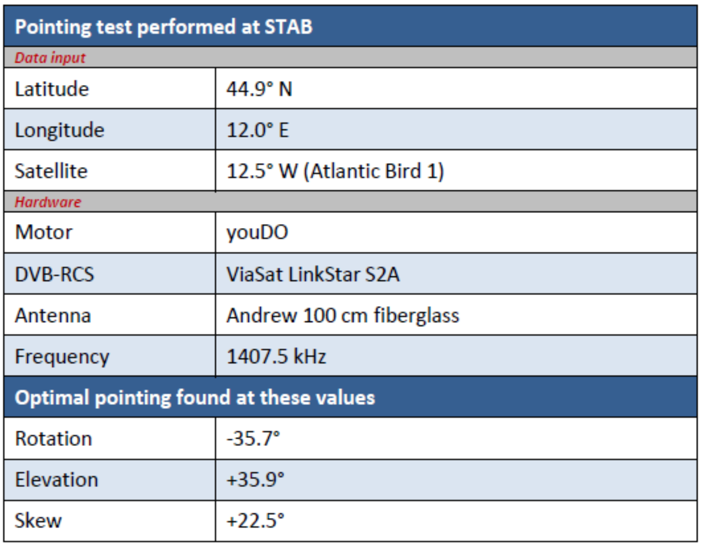
At 0,7 degrees misalignment, the 0,6 m terminal can still close easily the link, while the 0,45 m terminal has a margin slightly lower than 0 dB.

The requirements of the forward link alignment accuracy can be relaxed by introducing burst repetition. For example, it would be possible to tolerate 1 degree pointing accuracy for the 0,45 and 0,6 meters antenna transmitting the installation burst 7 and 15 times respectively.

9.6.2.2 Experimental results

This clause presents some experimental results regarding forward link alignment accuracy from [i.63]. The test configuration is shown in Table 9.9.

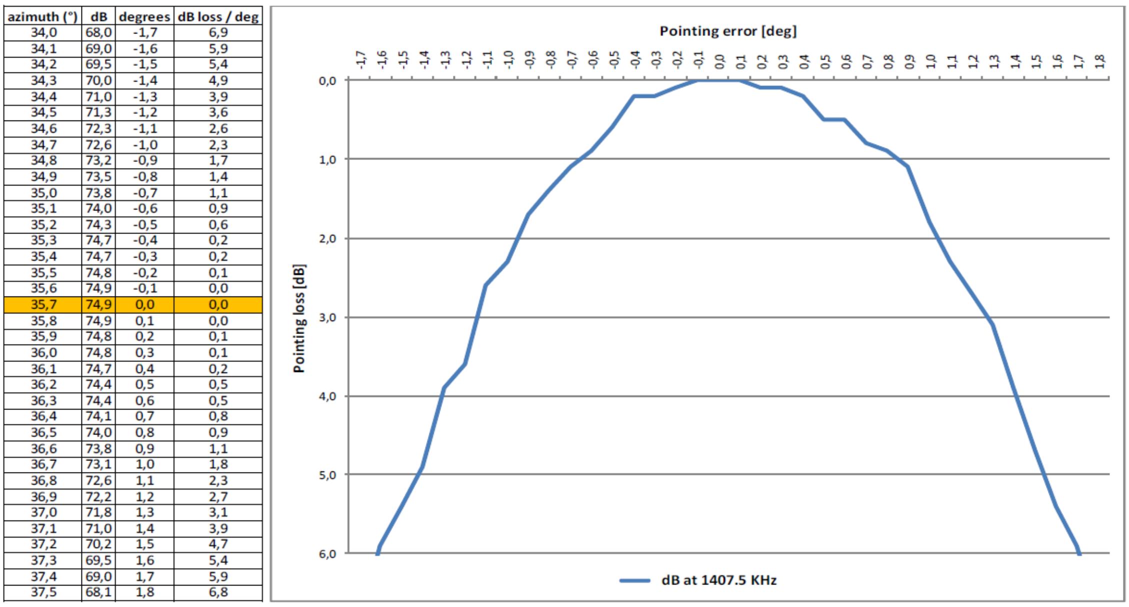
Table 9.9: Test Configuration for experimental result of forward link alignment accuracy



Pointing test performed at STAB	
Data input	
Latitude	44.9° N
Longitude	12.0° E
Satellite	12.5° W (Atlantic Bird 1)
Hardware	
Motor	youDO
DVB-RCS	ViaSat LinkStar S2A
Antenna	Andrew 100 cm fiberglass
Frequency	1407.5 kHz
Optimal pointing found at these values	
Rotation	-35.7°
Elevation	+35.9°
Skew	+22.5°

9.6.2.2.1 Pointing losses

The test motor is driven at 0,1-degree steps measuring the received signal strength at each step. Two sets of graphs are generated - one obtained from horizontal (azimuth) movements and the other from vertical (elevation) ones, starting from the position of optimum pointing.



azimuth (°)	dB	degrees	dB loss / deg
34,0	68,0	-1,7	6,9
34,1	69,0	-1,6	5,9
34,2	69,5	-1,5	5,4
34,3	70,0	-1,4	4,9
34,4	71,0	-1,3	3,9
34,5	71,3	-1,2	3,6
34,6	72,3	-1,1	2,6
34,7	72,6	-1,0	2,3
34,8	73,2	-0,9	1,7
34,9	73,5	-0,8	1,4
35,0	73,8	-0,7	1,1
35,1	74,0	-0,6	0,9
35,2	74,3	-0,5	0,6
35,3	74,7	-0,4	0,2
35,4	74,7	-0,3	0,2
35,5	74,8	-0,2	0,1
35,6	74,9	-0,1	0,0
35,7	74,9	0,0	0,0
35,8	74,9	0,1	0,0
35,9	74,8	0,2	0,1
36,0	74,8	0,3	0,1
36,1	74,7	0,4	0,2
36,2	74,4	0,5	0,5
36,3	74,4	0,6	0,5
36,4	74,1	0,7	0,8
36,5	74,0	0,8	0,9
36,6	73,8	0,9	1,1
36,7	73,1	1,0	1,8
36,8	72,6	1,1	2,3
36,9	72,2	1,2	2,7
37,0	71,8	1,3	3,1
37,1	71,0	1,4	3,9
37,2	70,2	1,5	4,7
37,3	69,5	1,6	5,4
37,4	69,0	1,7	5,9
37,5	68,1	1,8	6,8

Figure 9.17: Pointing loss as azimuth changes at 0,1-degree steps

elevation (°)	dB	degrees	dB loss / deg
34,3	68,5	-1,6	6,5
34,4	68,5	-1,5	6,5
34,5	68,5	-1,4	6,5
34,6	69,6	-1,3	5,4
34,7	70,3	-1,2	4,7
34,8	71,4	-1,1	3,6
34,9	71,9	-1,0	3,1
35,0	72,4	-0,9	2,6
35,1	73,2	-0,8	1,8
35,2	73,5	-0,7	1,5
35,3	73,9	-0,6	1,1
35,4	74,2	-0,5	0,8
35,5	74,5	-0,4	0,5
35,6	74,7	-0,3	0,3
35,7	74,9	-0,2	0,1
35,8	75,0	-0,1	0,0
35,9	75,0	0,0	0,0
36,0	75,0	0,1	0,0
36,1	74,9	0,2	0,1
36,2	74,9	0,3	0,1
36,3	74,8	0,4	0,2
36,4	74,7	0,5	0,3
36,5	74,3	0,6	0,7
36,6	74,0	0,7	1,0
36,7	73,7	0,8	1,3
36,8	73,3	0,9	1,7
36,9	73,0	1,0	2,0
37,0	72,5	1,1	2,5
37,1	71,6	1,2	3,4
37,2	71,0	1,3	4,0
37,3	70,0	1,4	5,0
37,4	69,3	1,5	5,7
37,5	68,3	1,6	6,7

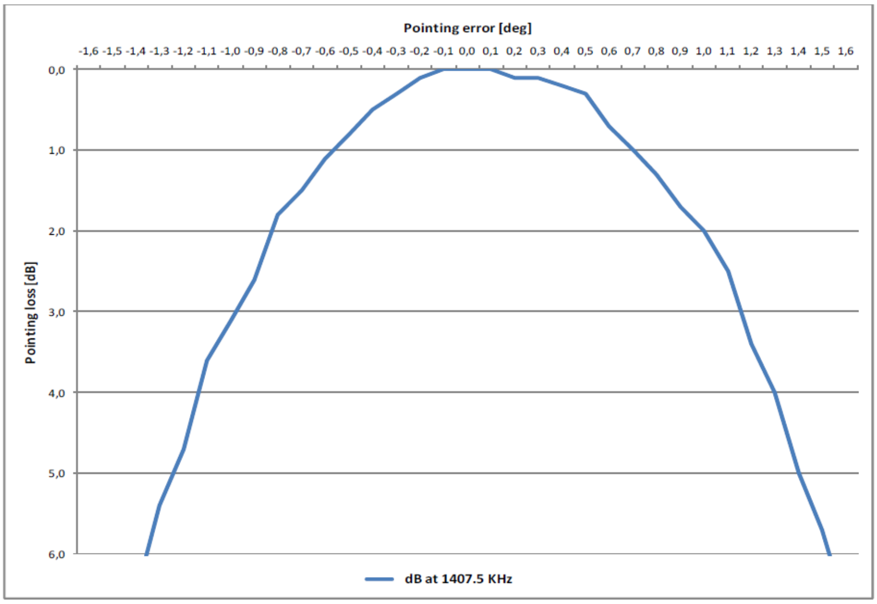
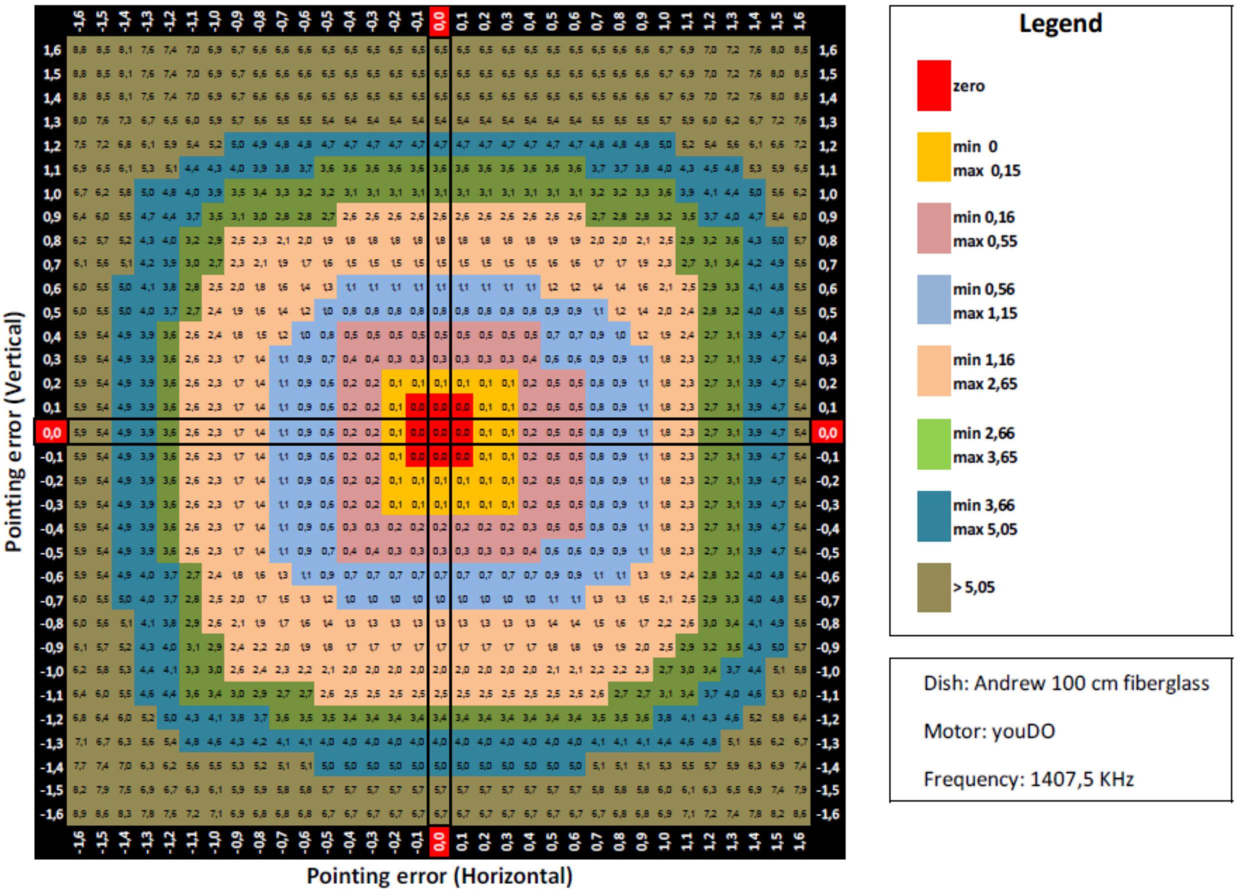


Figure 9.18: Pointing loss as elevation changes at 0,1-degree steps

The results shown in Figures 9.15 and 9.16 are combined into the 2-D plot in Figure 9.19. As shown, the point of minimum signal loss coincides with the centre of the "excellent zone".



	-1,6	-1,5	-1,4	-1,3	-1,2	-1,1	-1,0	-0,9	-0,8	-0,7	-0,6	-0,5	-0,4	-0,3	-0,2	-0,1	0,0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1,0	1,1	1,2	1,3	1,4	1,5	1,6	
1,6	8,8	8,5	8,1	7,6	7,4	7,0	6,9	6,7	6,6	6,6	6,6	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,6	6,6	6,7	6,9	7,0	7,2	7,6	8,0	8,5	1,6
1,5	8,8	8,5	8,1	7,6	7,4	7,0	6,9	6,7	6,6	6,6	6,6	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,6	6,6	6,7	6,9	7,0	7,2	7,6	8,0	8,5	1,5
1,4	8,8	8,5	8,1	7,6	7,4	7,0	6,9	6,7	6,6	6,6	6,6	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,5	6,6	6,6	6,7	6,9	7,0	7,2	7,6	8,0	8,5	1,4
1,3	8,0	7,6	7,3	6,7	6,5	6,0	5,9	5,7	5,6	5,5	5,5	5,4	5,4	5,4	5,4	5,4	5,4	5,4	5,4	5,4	5,4	5,4	5,4	5,5	5,5	5,5	5,7	5,9	6,0	6,2	6,7	7,2	7,6	1,3
1,2	7,5	7,2	6,8	6,1	5,9	5,4	5,2	5,0	4,9	4,8	4,8	4,7	4,7	4,7	4,7	4,7	4,7	4,7	4,7	4,7	4,7	4,7	4,7	4,8	4,8	4,8	5,0	5,2	5,4	5,6	6,1	6,6	7,2	1,2
1,1	6,9	6,5	6,1	5,3	5,1	4,4	4,3	4,0	3,9	3,8	3,7	3,6	3,6	3,6	3,6	3,6	3,6	3,6	3,6	3,6	3,6	3,6	3,6	3,7	3,7	3,8	4,0	4,3	4,5	4,8	5,3	5,9	6,5	1,1
1,0	6,7	6,2	5,8	5,0	4,8	4,0	3,9	3,5	3,4	3,3	3,2	3,2	3,1	3,1	3,1	3,1	3,1	3,1	3,1	3,1	3,1	3,1	3,1	3,2	3,2	3,3	3,6	3,9	4,1	4,4	5,0	5,6	6,2	1,0
0,9	6,4	6,0	5,5	4,7	4,4	3,7	3,5	3,1	3,0	2,8	2,8	2,7	2,6	2,6	2,6	2,6	2,6	2,6	2,6	2,6	2,6	2,6	2,6	2,7	2,8	2,8	3,2	3,5	3,7	4,0	4,7	5,4	6,0	0,9
0,8	6,2	5,7	5,2	4,3	4,0	3,2	2,9	2,5	2,3	2,1	2,0	1,9	1,8	1,8	1,8	1,8	1,8	1,8	1,8	1,8	1,8	1,9	1,9	2,0	2,0	2,1	2,5	2,9	3,2	3,6	4,3	5,0	5,7	0,8
0,7	6,1	5,6	5,1	4,2	3,9	3,0	2,7	2,3	2,1	1,9	1,7	1,6	1,5	1,5	1,5	1,5	1,5	1,5	1,5	1,5	1,5	1,6	1,6	1,7	1,7	1,9	2,3	2,7	3,1	3,4	4,2	4,9	5,6	0,7
0,6	6,0	5,5	5,0	4,1	3,8	2,8	2,5	2,0	1,8	1,6	1,4	1,3	1,1	1,1	1,1	1,1	1,1	1,1	1,1	1,1	1,1	1,2	1,2	1,4	1,4	1,6	2,1	2,5	2,9	3,3	4,1	4,8	5,5	0,6
0,5	6,0	5,5	5,0	4,0	3,7	2,7	2,4	1,9	1,6	1,4	1,2	1,0	0,8	0,8	0,8	0,8	0,8	0,8	0,8	0,8	0,8	0,9	0,9	1,1	1,2	1,4	2,0	2,4	2,8	3,2	4,0	4,8	5,5	0,5
0,4	5,9	5,4	4,9	3,9	3,6	2,6	2,4	1,8	1,5	1,2	1,0	0,8	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,7	0,7	0,9	1,0	1,2	1,9	2,4	2,7	3,1	3,9	4,7	5,4	0,4
0,3	5,9	5,4	4,9	3,9	3,6	2,6	2,3	1,7	1,4	1,1	0,9	0,7	0,4	0,4	0,3	0,3	0,3	0,3	0,3	0,3	0,4	0,6	0,6	0,9	0,9	1,1	1,8	2,3	2,7	3,1	3,9	4,7	5,4	0,3
0,2	5,9	5,4	4,9	3,9	3,6	2,6	2,3	1,7	1,4	1,1	0,9	0,6	0,2	0,2	0,1	0,1	0,1	0,1	0,1	0,1	0,2	0,5	0,5	0,8	0,9	1,1	1,8	2,3	2,7	3,1	3,9	4,7	5,4	0,2
0,1	5,9	5,4	4,9	3,9	3,6	2,6	2,3	1,7	1,4	1,1	0,9	0,6	0,2	0,2	0,1	0,0	0,0	0,0	0,1	0,1	0,2	0,5	0,5	0,8	0,9	1,1	1,8	2,3	2,7	3,1	3,9	4,7	5,4	0,1
0,0	5,9	5,4	4,9	3,9	3,6	2,6	2,3	1,7	1,4	1,1	0,9	0,6	0,2	0,2	0,1	0,0	0,0	0,0	0,1	0,1	0,2	0,5	0,5	0,8	0,9	1,1	1,8	2,3	2,7	3,1	3,9	4,7	5,4	0,0
-0,1	5,9	5,4	4,9	3,9	3,6	2,6	2,3	1,7	1,4	1,1	0,9	0,6	0,2	0,2	0,1	0,0	0,0	0,0	0,1	0,1	0,2	0,5	0,5	0,8	0,9	1,1	1,8	2,3	2,7	3,1	3,9	4,7	5,4	-0,1
-0,2	5,9	5,4	4,9	3,9	3,6	2,6	2,3	1,7	1,4	1,1	0,9	0,6	0,2	0,2	0,1	0,1	0,1	0,1	0,1	0,1	0,2	0,5	0,5	0,8	0,9	1,1	1,8	2,3	2,7	3,1	3,9	4,7	5,4	-0,2
-0,3	5,9	5,4	4,9	3,9	3,6	2,6	2,3	1,7	1,4	1,1	0,9	0,6	0,2	0,2	0,1	0,1	0,1	0,1	0,1	0,1	0,2	0,5	0,5	0,8	0,9	1,1	1,8	2,3	2,7	3,1	3,9	4,7	5,4	-0,3
-0,4	5,9	5,4	4,9	3,9	3,6	2,6	2,3	1,7	1,4	1,1	0,9	0,6	0,3	0,3	0,2	0,2	0,2	0,2	0,2	0,2	0,3	0,5	0,5	0,8	0,9	1,1	1,8	2,3	2,7	3,1	3,9	4,7	5,4	-0,4
-0,5	5,9	5,4	4,9	3,9	3,6	2,6	2,3	1,7	1,4	1,1	0,9	0,7	0,4	0,4	0,3	0,3	0,3	0,3	0,3	0,3	0,4	0,6	0,6	0,9	0,9	1,1	1,8	2,3	2,7	3,1	3,9	4,7	5,4	-0,5
-0,6	5,9	5,4	4,9	4,0	3,7	2,7	2,4	1,8	1,6	1,3	1,1	0,9	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,9	0,9	1,1	1,1	1,3	1,9	2,4	2,8	3,2	4,0	4,8	5,4	-0,6
-0,7	6,0	5,5	5,0	4,0	3,7	2,8	2,5	2,0	1,7	1,5	1,3	1,2	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,1	1,1	1,3	1,3	1,5	2,1	2,5	2,9	3,3	4,0	4,8	5,5	-0,7
-0,8	6,0	5,6	5,1	4,1	3,8	2,9	2,6	2,1	1,9	1,7	1,6	1,4	1,3	1,3	1,3	1,3	1,3	1,3	1,3	1,3	1,3	1,4	1,4	1,5	1,6	1,7	2,2	2,6	3,0	3,4	4,1	4,9	5,6	-0,8
-0,9	6,1	5,7	5,2	4,3	4,0	3,1	2,9	2,4	2,2	2,0	1,9	1,8	1,7	1,7	1,7	1,7	1,7	1,7	1,7	1,7	1,7	1,8	1,8	1,9	1,9	2,0	2,5	2,9	3,2	3,5	4,3	5,0	5,7	-0,9
-1,0	6,2	5,8	5,3	4,4	4,1	3,3	3,0	2,6	2,4	2,3	2,2	2,1	2,0	2,0	2,0	2,0	2,0	2,0	2,0	2,0	2,0	2,1	2,1	2,2	2,2	2,3	2,7	3,0	3,4	3,7	4,4	5,1	5,8	-1,0
-1,1	6,4	6,0	5,5	4,6	4,4	3,6	3,4	3,0	2,9	2,7	2,7	2,6	2,5	2,5	2,5	2,5	2,5	2,5	2,5	2,5	2,5	2,5	2,5	2,6	2,7	2,7	3,1	3,4	3,7	4,0	4,6	5,3	6,0	-1,1
-1,2	6,8	6,4	6,0	5,2	5,0	4,3	4,1	3,8	3,7	3,6	3,5	3,5	3,4	3,4	3,4	3,4	3,4	3,4	3,4	3,4	3,4	3,4	3,4	3,5	3,5	3,6	3,8	4,1	4,3	4,6	5,2	5,8	6,4	-1,2
-1,3	7,1	6,7	6,3	5,6	5,4	4,8	4,6	4,3	4,2	4,1	4,1	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,1	4,1	4,1	4,4	4,6	4,8	5,1	5,6	6,2	6,7	-1,3
-1,4	7,7	7,4	7,0	6,3	6,2	5,6	5,5	5,3	5,2	5,1	5,1	5,0	5,0	5,0	5,0	5,0	5,0	5,0	5,0	5,0	5,0	5,0	5,0	5,1	5,1	5,1	5,3	5,5	5,7	5,9	6,3	6,9	7,4	-1,4
-1,5	8,2	7,9	7,5	6,9	6,7	6,3	6,1	5,9	5,9	5,8	5,8	5,7	5,7	5,7	5,7	5,7	5,7	5,7	5,7	5,7	5,7	5,7	5,7	5,8	5,8	5,8	6,0	6,1	6,3	6,5	6,9	7,4	7,9	-1,5
-1,6	8,9	8,6	8,3	7,8	7,6	7,2	7,1	6,9	6,8	6,8	6,8	6,7	6,7	6,7	6,7	6,7	6,7	6,7	6,7	6,7	6,7	6,7	6,7	6,7	6,8	6,8	6,9	7,1	7,2	7,4	7,8	8,2	8,6	-1,6
	-1,6	-1,5	-1,4	-1,3	-1,2	-1,1	-1,0	-0,9	-0,8	-0,7	-0,6	-0,5	-0,4	-0,3	-0,2	-0,1	0,0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1,0	1,1	1,2	1,3	1,4	1,5	1,6	

Legend

🟥 zero

🟨 min 0
max 0,15

min 0,16
max 0,55

min 0,56
max 1,15

min 1,16
max 2,65

🟩 min 2,66
max 3,65

min 3,66
max 5,05

> 5,05

Dish: Andrew 100 cm fiberglass

Motor: youDO

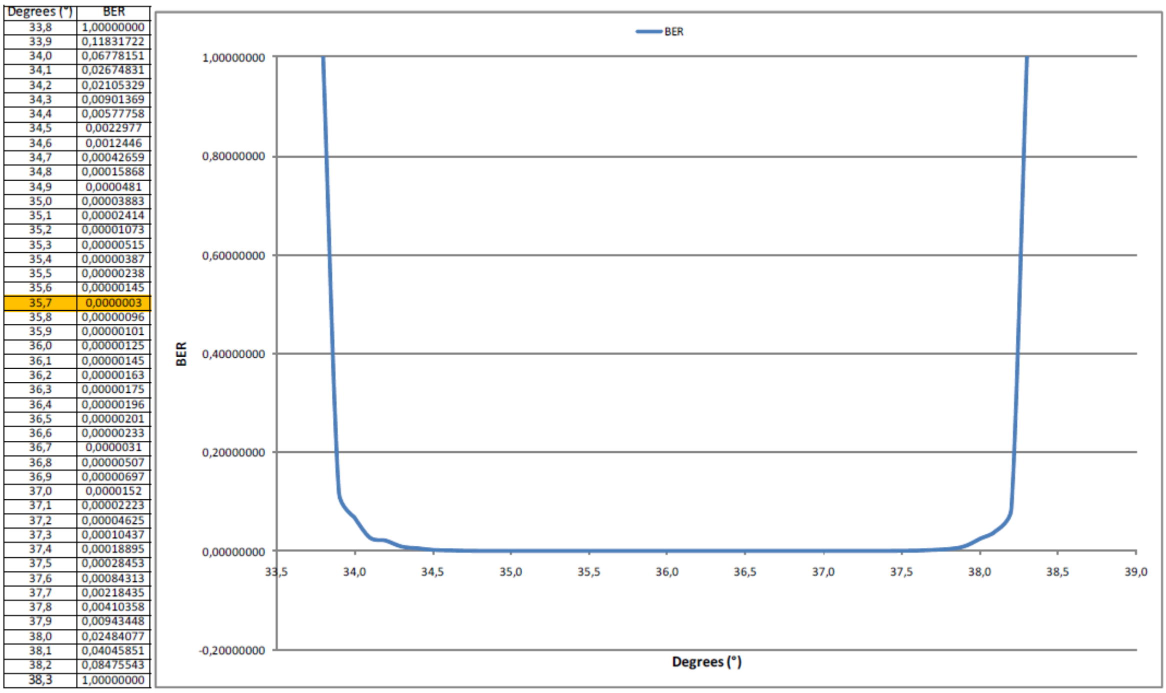
Frequency: 1407,5 KHz

Figure 9.19: Pointing loss (Horizontal: along azimuth) (Vertical: along elevation)

9.6.2.2.2 Bit Error Ratio Results

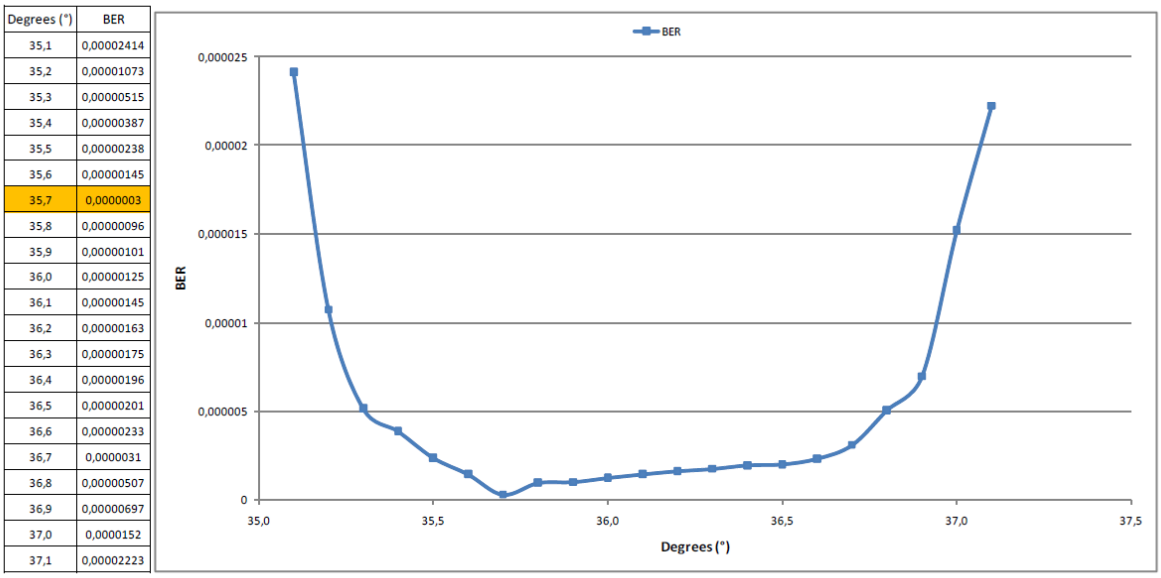
The main parameter to quantify the quality of the link is the BER: the number of received bits that have been altered due to noise, interference and distortion, divided by the total number of transferred bits during a studied time interval.

The test motor is driven at 0,1-degree steps measuring the received signal strength at each step. Two sets of graphs are generated - one obtained from horizontal (azimuth) movements and the other from vertical (elevation) ones, starting from the position of optimum pointing. BER measurements are collected over a 3-seconds observation interval for each step.



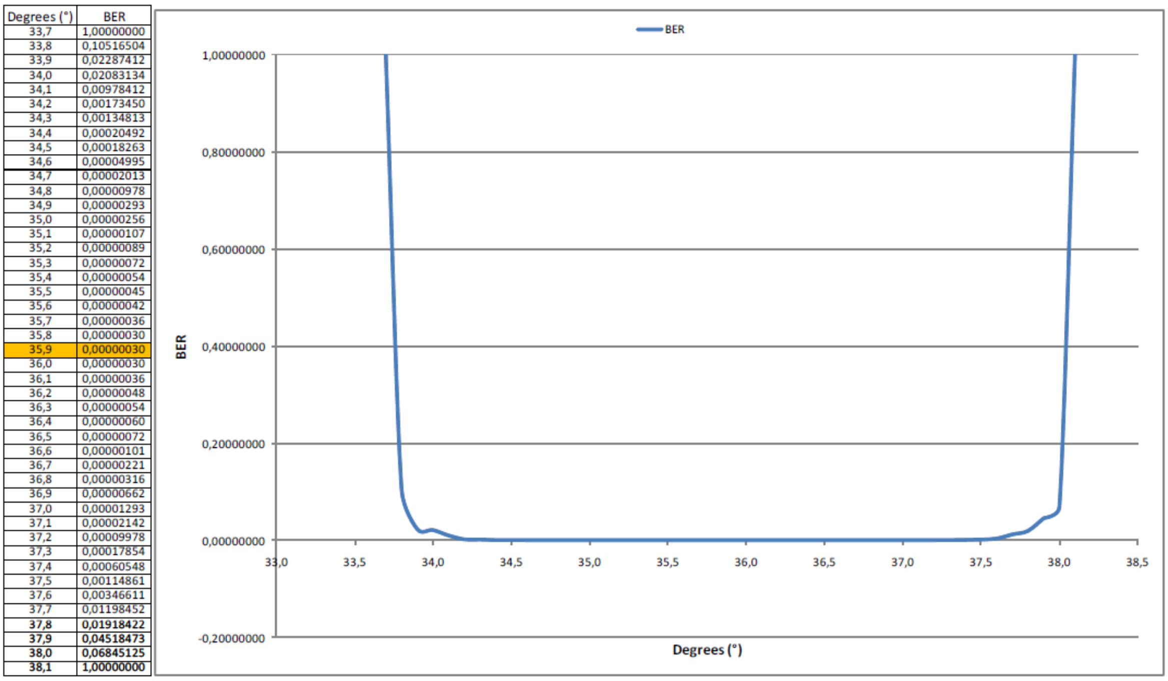
Degrees (°)	BER
33,8	1,00000000
33,9	0,11831722
34,0	0,06778151
34,1	0,02674831
34,2	0,02105329
34,3	0,00901369
34,4	0,00577758
34,5	0,0022977
34,6	0,0012446
34,7	0,00042659
34,8	0,00015868
34,9	0,0000481
35,0	0,00003883
35,1	0,00002414
35,2	0,00001073
35,3	0,00000515
35,4	0,00000387
35,5	0,00000238
35,6	0,00000145
35,7	0,0000003
35,8	0,00000096
35,9	0,00000101
36,0	0,00000125
36,1	0,00000145
36,2	0,00000163
36,3	0,00000175
36,4	0,00000196
36,5	0,00000201
36,6	0,00000233
36,7	0,0000031
36,8	0,00000507
36,9	0,00000697
37,0	0,0000152
37,1	0,00002223
37,2	0,00004625
37,3	0,00010437
37,4	0,00018895
37,5	0,00028453
37,6	0,00084313
37,7	0,00218435
37,8	0,00410358
37,9	0,00943448
38,0	0,02484077
38,1	0,04045851
38,2	0,08475543
38,3	1,00000000

Figure 9.20: BER as azimuth changes at 0,1-degree steps - Range: 33,8 - to - 38,3 degrees



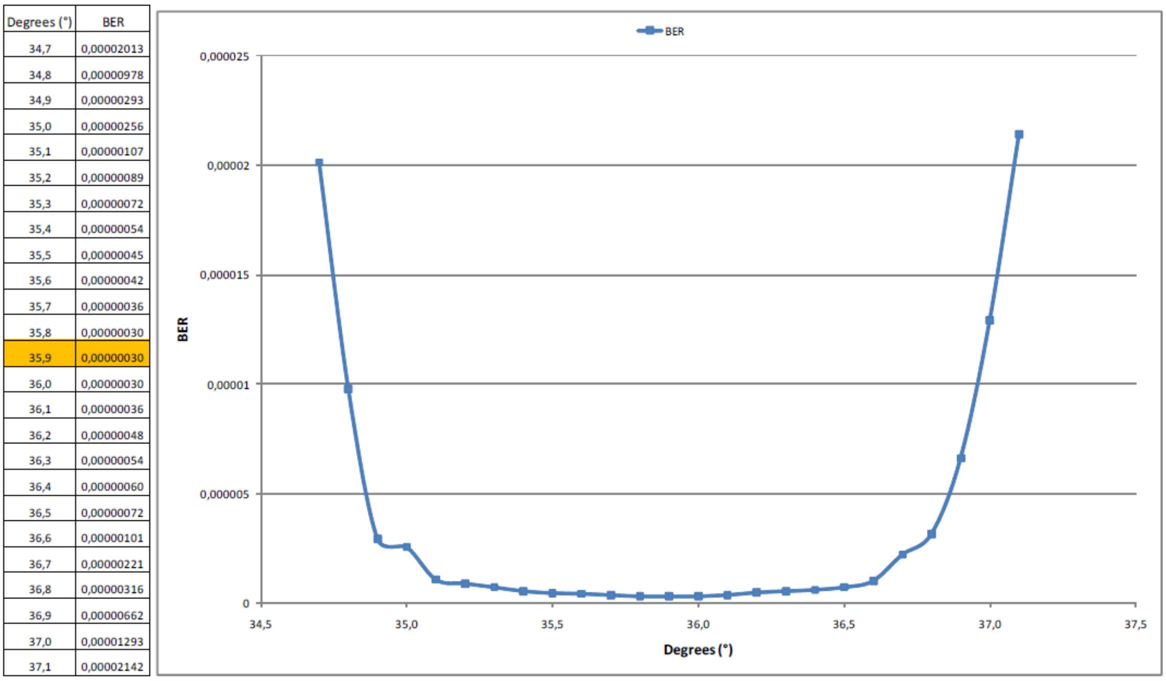
Degrees (°)	BER
35,1	0,00002414
35,2	0,00001073
35,3	0,00000515
35,4	0,00000387
35,5	0,00000238
35,6	0,00000145
35,7	0,0000003
35,8	0,00000096
35,9	0,00000101
36,0	0,00000125
36,1	0,00000145
36,2	0,00000163
36,3	0,00000175
36,4	0,00000196
36,5	0,00000201
36,6	0,00000233
36,7	0,0000031
36,8	0,00000507
36,9	0,00000697
37,0	0,0000152
37,1	0,00002223

Figure 9.21: BER as azimuth changes at 0,1-degree steps - Range: 35,1 - to - 37,1 degrees



Degrees (°)	BER
33,7	1,00000000
33,8	0,10516504
33,9	0,02287412
34,0	0,02083134
34,1	0,00978412
34,2	0,00173450
34,3	0,00134813
34,4	0,00020492
34,5	0,00018263
34,6	0,00004995
34,7	0,00002013
34,8	0,00000978
34,9	0,00000293
35,0	0,00000256
35,1	0,00000107
35,2	0,00000089
35,3	0,00000072
35,4	0,00000054
35,5	0,00000045
35,6	0,00000042
35,7	0,00000036
35,8	0,00000030
35,9	0,00000030
36,0	0,00000030
36,1	0,00000036
36,2	0,00000048
36,3	0,00000054
36,4	0,00000060
36,5	0,00000072
36,6	0,00000101
36,7	0,00000221
36,8	0,00000316
36,9	0,00000662
37,0	0,00001293
37,1	0,00002142
37,2	0,00009978
37,3	0,00017854
37,4	0,00060548
37,5	0,00114861
37,6	0,00346611
37,7	0,01198452
37,8	0,01918422
37,9	0,04518473
38,0	0,06845125
38,1	1,00000000

Figure 9.22: BER as elevation changes at 0,1-degree steps - Range: 33,7 - to - 38,1 degrees



Degrees (°)	BER
34,7	0,00002013
34,8	0,00000978
34,9	0,00000293
35,0	0,00000256
35,1	0,00000107
35,2	0,00000089
35,3	0,00000072
35,4	0,00000054
35,5	0,00000045
35,6	0,00000042
35,7	0,00000036
35,8	0,00000030
35,9	0,00000030
36,0	0,00000030
36,1	0,00000036
36,2	0,00000048
36,3	0,00000054
36,4	0,00000060
36,5	0,00000072
36,6	0,00000101
36,7	0,00000221
36,8	0,00000316
36,9	0,00000662
37,0	0,00001293
37,1	0,00002142

Figure 9.23: BER as elevation changes at 0,1-degree steps - Range: 35,7 - to - 37,1 degrees

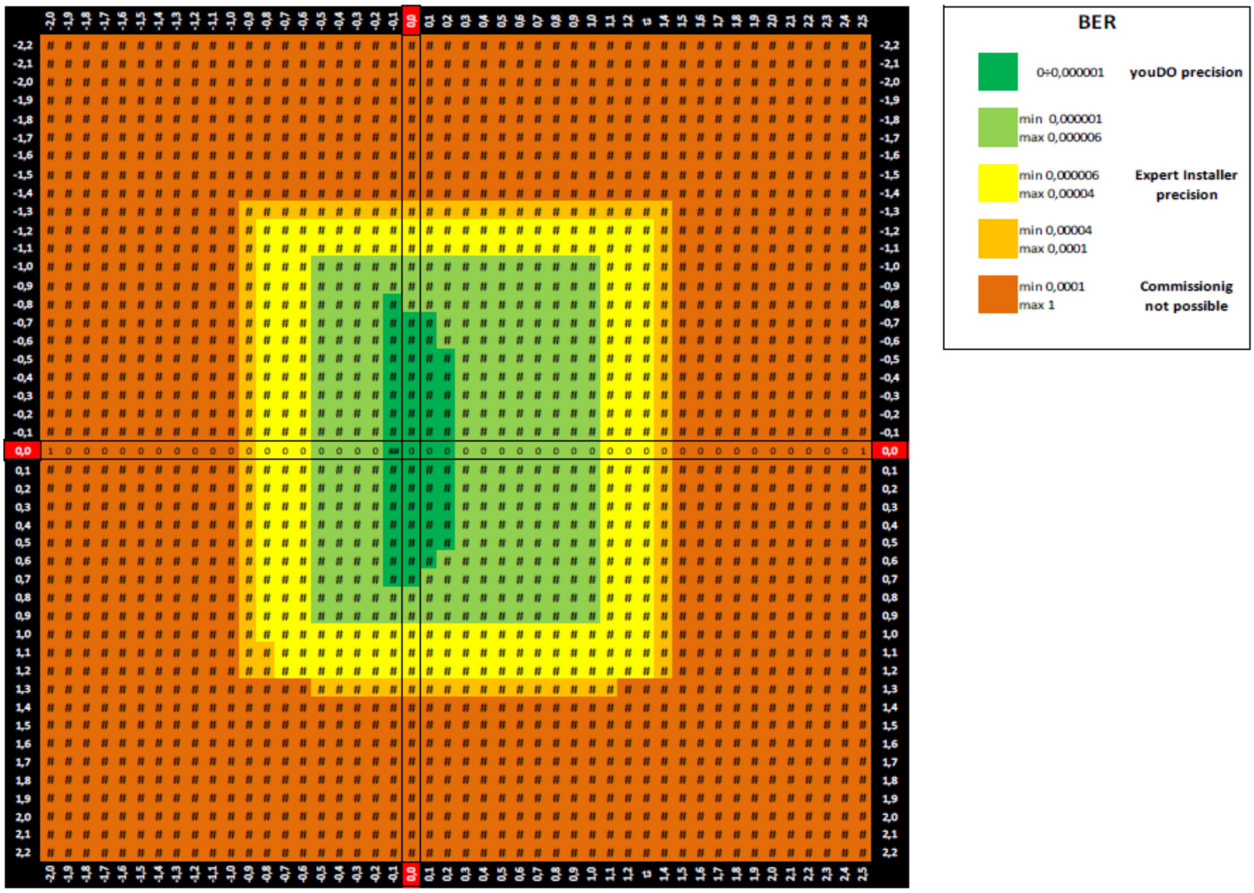


Figure 9.24: BER variation along azimuth (horizontal) and elevation (vertical)

Figure 9.24 shows that the point of minimum BER level coincides with the centre of the "excellent zone".

Figure 9.25 shows the impact of polarization angle - with linear polarization - on the BER. After the forward link alignment is achieved, the polarizer is rotated at 1-degree steps. The BER is measured over 3-second intervals at each step.

Degrees (°)	BER
9,5	0,00002270
10,5	0,00001620
11,5	0,00000750
12,5	0,00000550
13,5	0,00000380
14,5	0,00000260
15,5	0,00000190
16,5	0,00000130
17,5	0,00000100
18,5	0,00000100
19,5	0,00000080
20,5	0,00000060
21,5	0,00000040
22,5	0,00000030
23,5	0,00000040
24,5	0,00000050
25,5	0,00000060
26,5	0,00000060
27,5	0,00000070
28,5	0,00000090
29,5	0,00000110
30,5	0,00000170
31,5	0,00000250
32,5	0,00000400
33,5	0,00000550
34,5	0,00000750
35,5	0,00001300
36,5	0,00002000

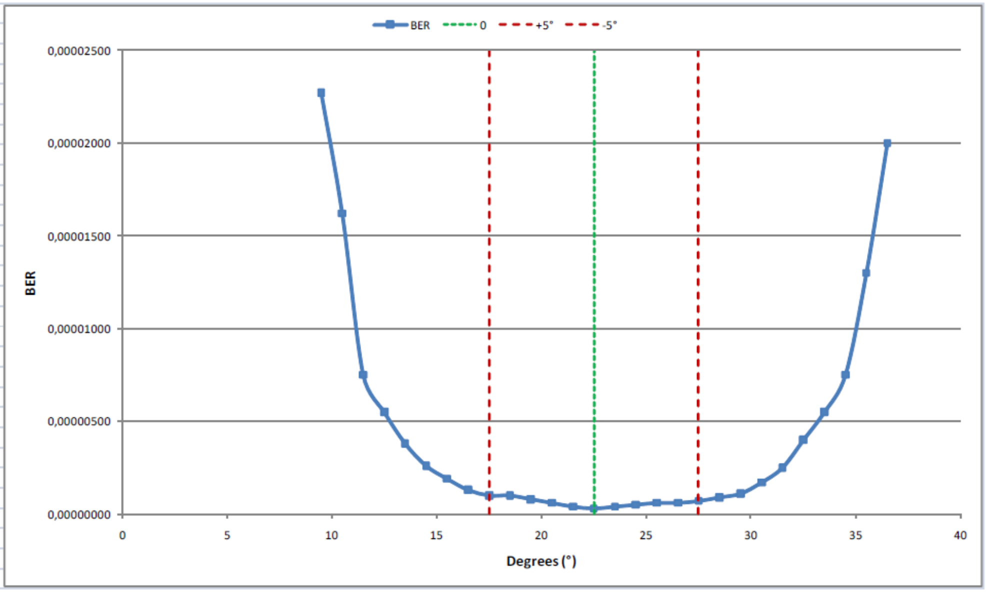


Figure 9.25: BER variation along skew [-5, +5] degrees

10 System Implementation Guidelines

Examples of transmit/receive performance characteristics are provided to be used as guidelines in system performance evaluation, dimensioning and network capacity assessment.

10.1 Typical RCST RF Signal Characteristics

This clause describes a set of typical RCST performance characteristics to be used as guidelines for evaluating the overall system performance.

10.1.1 Phase Noise

Phase noise characteristics are typically dominated by the phase noise produced by the IDU/ODU. Table 10.1 presents typical phase noise characteristics of the RCST.

In order to evaluate the overall link performance, it is assumed that the combined effect of all other phase noise sources in the link is at least 10 dB lower than the typical values presented here.

Table 10.1: RCST Transmit Phase Noise

Item	Description	Overall RCST	Remarks
A	SSB Phase Noise (for Symbol rate ≥ 128 kBaud) 10 Hz 100 Hz 1 kHz 10 kHz 100 kHz > 1 MHz	 ≤ -16 dBc/Hz ≤ -54 dBc/Hz ≤ -64 dBc/Hz ≤ -74 dBc/Hz ≤ -89 dBc/Hz ≤ -106 dBc/Hz	It is assumed that the combined effect of other phase noise sources in the transmission path (hub and satellite included) is at least 10 dB better. The split of SSB phase noise between IDU and ODU is left to the manufacturers' decision.
B	SSB Phase Noise (for 128 kBaud > Symbol rate ≥ 8 kBaud) 10 Hz 100 Hz 1 kHz 10 kHz 100 kHz > 1 MHz	 ≤ -30 dBc/Hz ≤ -60 dBc/Hz ≤ -70 dBc/Hz ≤ -74 dBc/Hz ≤ -89 dBc/Hz ≤ -106 dBc/Hz	

10.1.2 Carrier Frequency Accuracy

Clause 7.3.9.3 of the normative document [i.1] defines the requirement on the carrier frequency accuracy of the RCST as 10^{-8} (root mean square) relative to the nominal carrier frequency. The corresponding maximum error value should be 6×10^{-8} relative to the nominal carrier frequency.

The frequency accuracy of the terminal burst is the result of a number of contributors. Typical fixed contribution - i.e. the frequency accuracy typical of a classical system - is provided in Table 10.2.

Table 10.2: Carrier Frequency Accuracy

Contributor	Value	Application	Source	Ku-band (note 1) (Hz)	Ka-band (note 2) (Hz)
Terminal Frequency Accuracy	6 × 10^{-8}	U/L	Worst case	870 Hz	1 800 Hz
Gateway Frequency Accuracy	10^{-8}	D/L	Typical	128 Hz	202 Hz
Satellite Frequency Accuracy	10^{-7}	delta (D/L, U/L)	Typical	175 Hz	980 Hz
Satellite motion (Doppler effect)	10^{-8} (note 3)	U/L + D/L	Typical	418 Hz	802 Hz
Total contribution				1 590 Hz	3 784 Hz
NOTE 1: Ku-band: 14,5 GHz uplink, 12,75 GHz downlink. NOTE 2: Ka-band: 30,0 GHz uplink, 20,2 GHz downlink. NOTE 3: Dependent on station keeping strategy, could be improved to typically 5 x 10^{-9}.					

Terminal frequency accuracy: this value corresponds to the maximum error value of the RCST normalized frequency accuracy (considering to 6 times the RMS value). This value excludes Doppler shift and should be considered as normalized with respect to master synchronization reference at the NCC.

Gateway frequency accuracy: typical value for the gateway receiver (ODU+IDU) frequency accuracy.

Satellite Frequency Accuracy: typical value for the satellite uplink and downlink frequency accuracy. For transparent satellite, the resulting effect of frequency accuracy is computed on the difference between uplink and downlink frequency.

Doppler Effect due to satellite motion: typical value for the satellite Doppler shift. This value depends on the station-keeping strategy. This effect results in two contributions on the return path from the terminal to the gateway: 1) offset in RCST transmit frequency due to Doppler shift on the forward path, and 2) Frequency offset due to Doppler shift on the return path. The first contribution is a consequence of locking the terminal local frequency to the transmit PCR reference which has induced Doppler (NCR time drift). This would cause a frequency offset on the terminal transmit frequency. The second contribution is the classical Doppler frequency offset due to satellite motion, which has to be accounted for on the uplink (from terminal to satellite) but also on downlink (from satellite to gateway).

10.1.3 Amplitude Variation

Table 10.3 provides peak-to-peak limits of the carrier output amplitude variations, assuming a continuous wave transmit signal, for different frequency ranges.

Table 10.3: Amplitude Variation

Amplitude Variation	Overall RCST	Remarks
In any 3 MHz band In any 20 MHz band In any 40 MHz band	< 0,5 dB p-p < 1,5 dB p-p < 2,0 dB p-p	The split of amplitude variation between IDU and ODU is left to the manufacturers' decision

10.1.4 I/Q Imbalance

The I/Q amplitude imbalance is typically less than ±0,5 dB p-p. I/Q quadrature imbalance is typically less than ±1 degree p-p. The I/Q amplitude offset is typically lower than 0,5 dB.

The maximum misalignment between I and Q symbols is typically less than 5 % of the symbol period.

10.1.5 Spurious Levels

Within the transmit band, the RCST should meet the spurious radiation such that for each spurious signal that it transmits outside the nominated bandwidth, the total EIRP of each spurious signal should not exceed a level of 60 dB below the total EIRP of the transmitted carrier (modulated or unmodulated). Within the transmit band, the transmission enabled RCST should not generate a noise EIRP density (dBW/Hz) exceeding:

Nominal EIRP (dBW) - 122 dBHz

outside the nominated bandwidth. In the transmission disabled state the limits is 30 dB more stringent.

For outside the transmit band, the requirements governing the RCST operation are specified in EN 301 459 [i.6] and EN 301 428 [i.7].

10.1.6 Non-linearity

The non-linear distortion impacting the RCST transmit RF signal is mainly caused by the high power amplifier at the ODU. Non-linear characteristics of the amplifier depends on several factors, including the technology (i.e. TWTA vs. SSPA), frequency band (e.g. C, Ku or Ka), transmit waveform (modulation scheme and pulse shaping filter), amplifier operating point (input signal level), environment conditions (e.g. temperature) as well as the device ageing.

The amplitude and phase characteristics of a non-linear device is typically provided as the power transfer (AM/AM) and phase transfer (AM/PM) functions that relate the output signal instantaneous power and phase to the input power. Typically, such characteristics are provided based on single-tone measurements of unmodulated carriers.

10.1.6.1 Power Transfer Characteristics

Figure 10.1 presents examples of power transfer of an SSPA. As shown in this figure, depending on the input signal power, the amplifier operates linearly at low input power (1 dB/dB gain), saturates at a high input power. In a mid-power range (depending on the device), the input signal power could be compressed or expanded.

Simulations involving nonlinearities usually require computation of the transfer characteristic at arbitrary points. Although measured date along with the interpolation between points could be used for such computation, the slope discontinuities introduced by nearly all interpolation techniques such as splines or polynomial techniques introduce artificial signal distortion, which could appear for example as increased side lobe levels or signal distortion.

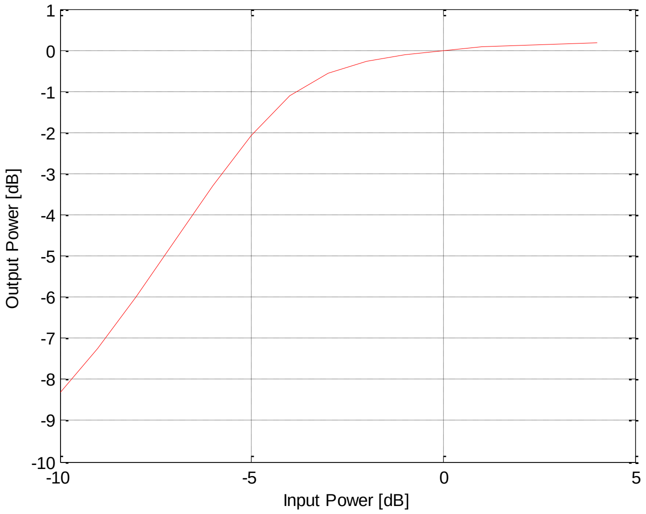


Figure 10.1: Example AM/AM non-linearity characteristics (measured data)

A mathematical expression that is commonly used to model SSPA's AM/AM characteristics has been described in [i.38]. Similar expression is presented below:

$$G(A) = \frac{gA}{\left(1 + \left(\frac{gA}{A_{sat}}\right)^{2s}\right)^{\frac{1}{2s}}}$$

where s is the smoothness factor, A_{sat} is the saturation amplitude and g defines the gain of the amplifier in the linear region. This model can be applied to modulated signals as a memoryless function to compute the instantaneous signal amplitude values. An example is presented in Figure 10.2, with s = 6, g = 1,122 and A_{sat} = 1,0351. Asymptotically, the model can present an ideal clipping law (for large values of s).

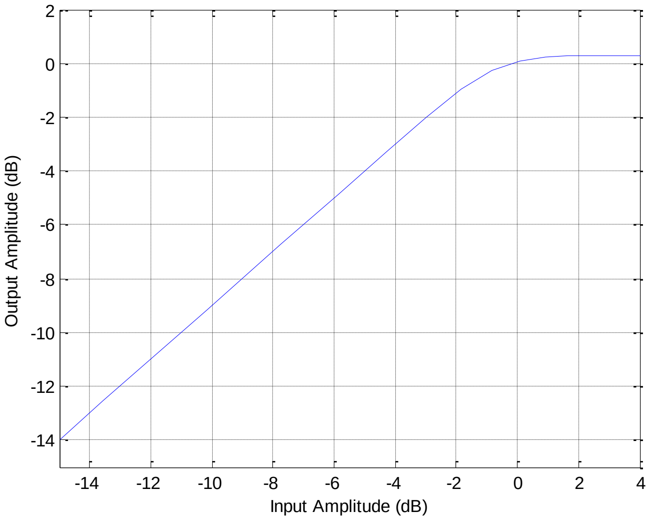


Figure 10.2: An example of AM/AM non-linearity according to Rapp Model [i.38] (with *s* = 6, *g* = 1,122 and A_{sat} = 1,0351)

The impact of the non-linear distortion on the overall system performance can be categorized as:

- **Power Saturation:** The output power does not proportionally increase with respect to the input power. This is reflected in the AM/AM characteristics of the amplifier.
- **Modulation Dependent Power Loss:** Depending on the modulation scheme, the output power loss can vary (compared to continuous wave, there is a higher output power loss for linearly modulated signals).
- **Out-of-Band power (spectral regrowth):** varies depending on the operating point and the input waveform characteristics.
- **In-band distortion:** The non-linearity will impact the signal characteristics. The impact can be seen as in-band interfering components, transmitted together with the original signal (inter-modulation interference).

10.1.6.2 Phase Transfer Characteristics

The phase transfer (AM/PM) characteristic of SSPA's is usually more benign than that found in TWT amplifiers. In the open literature, the impact of AM/PM conversion of SSPA on the RF signal is often considered to be negligible. There are however examples of measured data that indicate otherwise [i.44]. Examples of phase transfer models have also been provided in [i.44]. For DVB-RCS2 systems that can utilize higher order modulations particularly 16 QAM, it is recommended that the phase transfer characteristics of the non-linear device to be examined and to be considered in system performance evaluation.

It should be noted that there are pre-compensation techniques to be deployed at the transmitter to reduce the impact of such phase (and amplitude) distortions.

10.1.6.3 Power Amplifier Operating Point Stability

In order to maximum the efficiency of the power amplifier, it is desirable to minimize the output back-off. However, the actual operating point of the amplifier and the level OBO can vary due to several reasons, such as:

- Power Detector Instabilities (assuming a power detector used in a closed feedback loop, as shown in Figure 10.3):
  - Temperature stability
  - Detector frequency response: production test limit
  - Detector response to different modulations and symbol rates

- IBO instabilities:
    - Cable amplitude variation over frequency (e.g. 500 MHz frequency range)
    - IBO granularity (due to the driver settings)
- SSPA-related instabilities (ageing, temperature, etc.):
    - Saturation power
    - AM/AM and AM/PM curve shape

For linear modulation schemes, a closed loop control could be utilized in order to reduce the operating point instability when using. An example of such feedback control loop is shown schematically in Figure 10.3. As shown, in this figure, the transmit power level at the out of the power amplifier (SSPA) is measures (power detector) and the measurement results are used (in this example, power measurement values are communicated to the IDU via DISEqC^{TM} protocol [i.20]).

The RCST should indicate to the NCC the relative OBO used between QPSK and 8PSK, and between QPSK and 16QAM. The RCST will enforce this OBO autonomously in order to avoid violation of the PSD mask. The relative OBO indicated and enforced by the RCST is dependent on the local minimum OBO configured for the specific installation and the characteristics of the HPA.

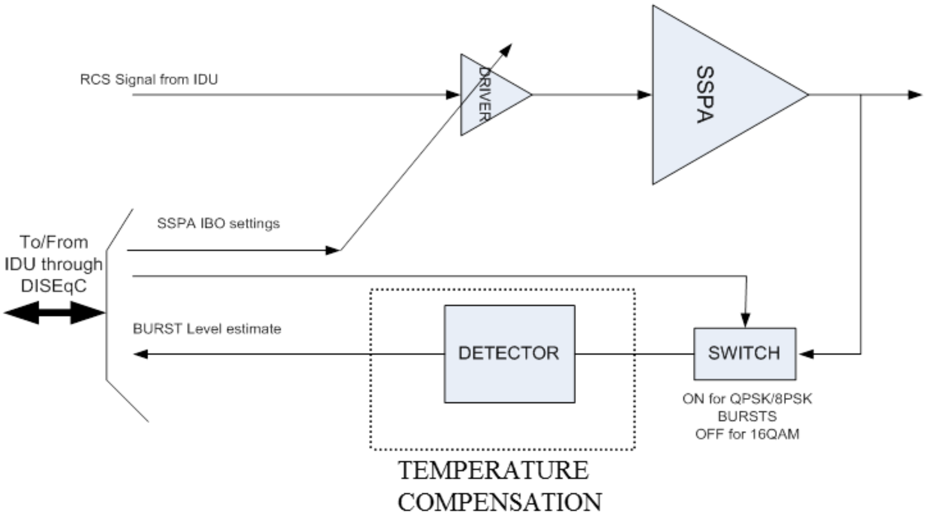


Figure 10.3: Feedback Loop for IDU/ODU Operating Point Power Stability

10.2 Receiver Performance

10.2.1 Linear Modulation

Turbo Decoder performance results in AWGN channel are presented in tables below. Results are based on software simulations with the following parameters:

- Number of Turbo Decoder Iterations: up to 8 iterations,
- Decoder Architecture: BCJR, Normalized Max-log-map,
- Synchronization: Ideal.

Table 10.4: AWGN Performance results for control bursts

Waveform ID Table A-1 of [i.1]	Burst Length	Modulation	Code rate	E_s/N_0 (dB) @ PER =				
				10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
2	262	QPSK	1/3	-0,53	0,06	0,50	0,87	1,29
41	3 236	BPSK	1/3	-4,00	-3,65	-3,38	-3,17	-2,98

Table 10.5: AWGN Performance results for Short Bursts (536 symbols)

Waveform ID Table A-1 of [i.1]	Burst Length	Modulation	Code rate	E_s/N_0 (dB) @ PER = 10^{-3}	E_s/N_0 (dB) @ PER = 10^{-5}
3	536	QPSK	1/3	-0,27	0,22
4	536	QPSK	1/2	1,92	2,34
5	536	QPSK	2/3	3,90	4,29
6	536	QPSK	3/4	4,93	5,36
7	536	QPSK	5/6	6,11	6,68
8	536	8PSK	2/3	7,71	8,08
9	536	8PSK	3/4	8,90	9,31
10	536	8PSK	5/6	10,43	10,85
11	536	16QAM	3/4	10,83	11,17
12	536	16QAM	5/6	12,16	12,56

Table 10.6: AWGN Performance results for Long Bursts (1616 symbols)

Waveform ID Table A-1 of [i.1]	Burst Length	Modulation	Code rate	E_s/N_0 (dB) @ PER = 10^{-3}	E_s/N_0 (dB) @ PER = 10^{-5}
13	1 616	QPSK	1/3	-0,80	-0,51
14	1 616	QPSK	1/2	1,49	1,71
15	1 616	QPSK	2/3	3,46	3,69
16	1 616	QPSK	3/4	4,50	4,73
17	1 616	QPSK	5/6	5,64	5,94
18	1 616	8PSK	2/3	7,29	7,49
19	1 616	8PSK	3/4	8,56	8,77
20	1 616	8PSK	5/6	10,02	10,23
21	1 616	16QAM	3/4	10,55	10,72
22	1 616	16QAM	5/6	11,86	12,04

Table 10.7: AWGN Performance results for Very Short Bursts (266 symbols)

Waveform ID Table A-1 of [i.1]	Burst Length	Modulation	Code rate	E_s/N_0 (dB) @ PER = 10^{-3}	E_s/N_0 (dB) @ PER = 10^{-5}
44	266	QPSK	5/6	6,52	7,3
45	266	8-PSK	2/3	8,2	8,71
46	266	8-PSK	3/4	9,41	10,04
47	266	8-PSK	5/6	10,83	11,59
48	266	16-QAM	3/4	11,24	11,73
49	266	16-QAM	5/6	12,56	13,18

Table 10.8: AWGN Performance results for BPSK Modulated Bursts (very long bursts)

Waveform ID Table A-1 of [i.1]	Burst Length	Modulation	Code rate	E_s/N_0 (dB) @ PER = 10^{-3}	E_s/N_0 (dB) @ PER = 10^{-5}
42	3 236	BPSK	1/3	-3,81	-3,52
43	3 236	BPSK	1/2	-1,53	-1,30

A summary of Spectral Efficiency for different waveforms are shown in Table 10.9 where the impact of the unique words, pilot symbols and guard symbols are included in the computation of the efficiency for each waveform. Reference burst waveforms, as listed in Table 10.9, are designed to fit all into a unified timeslot grid comprising 270 Physical layer symbols. The use of constant burst size can simplify RRM, ACM and payload size adaptation.

The RCST HPA saturation effects may require an RCST to reduce its emitted RF power (i.e. increasing the OBO) when utilizing higher order modulation schemes, thus requiring a higher link margin. The RCST may indicate the level of such RF power reduction to the NCC so the most appropriate waveform can be selected (as described in clause 7.1.1). The NCC may assume a worst case RF power reduction scenario if this information is not provided by the RCST.

Figure 10.4 illustrates the bit/symbol efficiency as a function of E_s/N_0 for 28 reference bursts (as per Table 10.9). A target packet error ratio of PER = 10^{-5} has been considered.

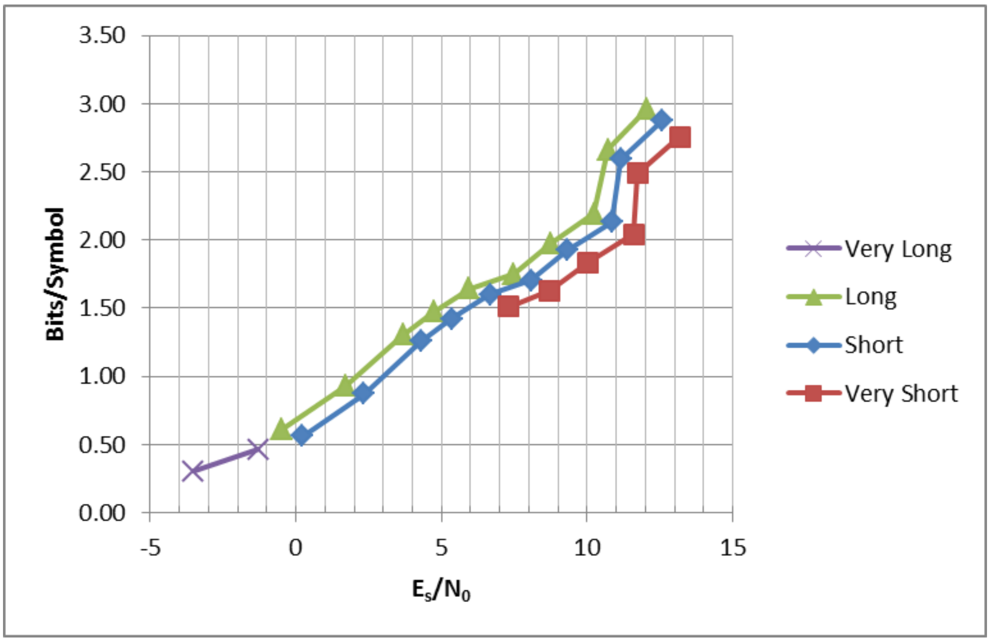


Figure 10.4: Efficiency versus sensitivity thresholds for 28 reference burst waveforms of 4 distinct sizes, at ideal synchronization in an AWGN channel

Table 10.9: Linear Modulation Waveforms Efficiency in AWGN Channel

Waveform ID (Table A-1 of [i.1])	Burst Size (symbols)	Guard (symbols)	Payload (bits)	Efficiency (Bits/Symbol)	E_s/N_0 @ PER = 10^{-5}
44	266	4	408	1,51	7,3
45	266	4	440	1,63	8,71
46	266	4	496	1,84	10,04
47	266	4	552	2,04	11,59
48	266	4	672	2,49	11,73
49	266	4	744	2,76	13,18
3	536	4	304	0,56	0,22
4	536	4	472	0,87	2,34
5	536	4	680	1,26	4,29
6	536	4	768	1,42	5,36
7	536	4	864	1,60	6,68
8	536	4	920	1,70	8,08
9	536	4	1 040	1,93	9,31
10	536	4	1 152	2,13	10,85
11	536	4	1 400	2,59	11,17
12	536	4	1 552	2,87	12,56
13	1 616	4	984	0,61	-0,51
14	1 616	4	1 504	0,93	1,71
15	1 616	4	2 112	1,30	3,69
16	1 616	4	2 384	1,47	4,73
17	1 616	4	2 664	1,64	5,94
18	1 616	4	2 840	1,75	7,49
19	1 616	4	3 200	1,98	8,77
20	1 616	4	3 552	2,19	10,23
21	1 616	4	4 312	2,66	10,72
22	1 616	4	4 792	2,96	12,04
42	3 236	4	984	0,30	-3,52
43	3 236	4	1 504	0,46	-1,3

10.2.2 CPM

10.2.2.1 Simulation Model

The CC-CPM simulation block diagram used for the performance assessment is shown in Figure 10.5.

The signal from the modulator is transmitted through the channel such that the signal at its output is given by:

$$r(t) = s(t) + n(t) + i(t),$$

where

- *n (t)* is AWGN, with single-sided power spectral density of N_0 (Watt/Hz).
- *i(t)* denotes interference (ACI) from four equally spaced carriers (i.e. two carriers on either side of the desired carrier s(t)).

If f_i $\mathrm{f_i}$and f_j $\mathrm{f_j}$are the centre frequencies for the i^{th} and j^{th} carriers, then the frequency separation between two immediate neighbors is, $\Delta_f = | f_i - f_j |$$|\Delta_f = |f_i - f_j|$, for all, $| i - j | = 1$$|i - j| = 1$. The spectral efficiency is measured as, $\eta = \dfrac{R_c \log_2 M}{\Delta_f T_s}$ and T_s is the symbol duration$\eta = \frac{R_c \times \log_2 M}{\Delta_f T}$. The carriers are homogenous, in that they all use the same CPM modulation parameters. The interfering carriers are assumed to be 3dB stronger than the desired carrier.

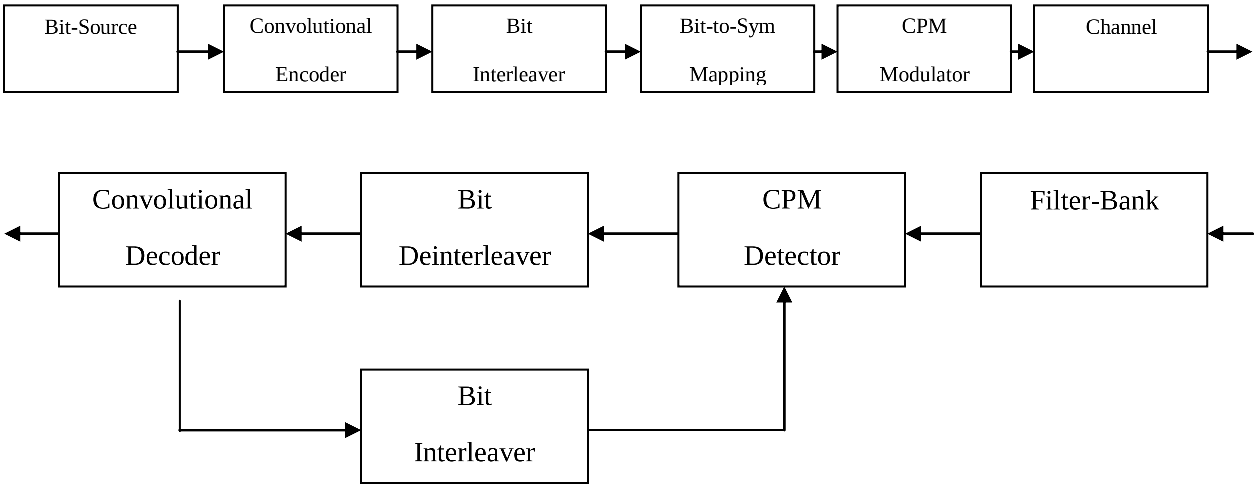


Figure 10.5: Block diagram of CPM transmitter, channel and the receiver

The block diagram shown above depicts the receiver blocks. A perfect timing recovery and carrier phase synchronization is assumed at the receiver. Additionally, no information is exchanged with the adjacent carriers' receivers. A low-pass filter can be applied at the receiver front-end to mitigate the ACI at higher spectral efficiencies. The CPM demodulator consists of a front-end filter-bank followed by the soft-in, soft-out (SISO) CPM detector. The CPM detector generates the maximum a posteriori (MAP) probabilities for the transmitted codebits using a trellis [i.39] or factor graph [i.40] describing the modulation.

The convolutional decoder also performs SISO detection by executing the BCJR algorithm [i.41] on the trellis describing the convolutional code. We note that the constraint length 3 convolutional code decodes on a 4 state trellis, whereas the constraint length 4 code requires an 8 state trellis. A maximum of 30 iterations are performed between the CPM SISO detector and the SISO convolutional decoder, during which extrinsic information is exchanged between them. In summary:

- Five carriers with identical channel code and CPM parameters are simulated
- Two interfering carriers on either side of desired carrier, each is 3 dB stronger than desired carrier
- No multi-user detection/ ACI cancellation is applied
- An ideal synchronization is assumed (no carrier frequency offset or phase noise is present)
- 30 iterations between the CPM detector and the convolutional decoder

10.2.2.2 Performance Results for CPM Waveforms

Performance results for three information block sizes and different spectral efficiencies are summarized in the following tables. Results are obtained using software simulation models described in clause 10.2.2.1 Waveform parameters as summarized tables below are in line with those identified in the normative document (See table A-2 of [i.1]).

It should be noted that the impact of known symbols, trellis termination symbols and the guard time are not taken into account in the computation of the spectral efficiency and the performance threshold values.

Table 10.10: Simulation Results for bursts containing 400 information bits

Waveform ID, Table A-2 of [i.1]	Spectral efficiency b/s/Hz	Modulation Index *h*	Pulse shape AV	Code rate	FEC Conv. Code	Normalized Carrier Spacing (*fT*)	Eb/No(dB) @ PER = 10^{-3}	Eb/No(dB)@ PER = 10^{-5}
3	0,5	2/5	α_{RC} = 0,98	0,5	$(5,7)_{octal}$	2,000	2,2	2,75
4	0,75	1/3	α_{RC} = 0,75	0,5	$(5,7)_{octal}$	1,333	2,75	3,25
5	1,10	2/7	α_{RC} = 0,75	2/3	$(5,7)_{octal}$	1,210	3,7	4,4
6	1,25	2/7	α_{RC} = 0,75	2/3	$(5,7)_{octal}$	1,067	4,4	5,2
7	1,50	1/4	α_{RC} = 0,75	4/5	$(15,17)_{octal}$	1,067	6,1	7,2
8	1,80	1/5	α_{RC} = 0,625	6/7	$(15,17)_{octal}$	0,974	9,2	11,1

Table 10.11: Simulation Results for bursts containing 1 024 information bits

Waveform ID, Table A-2 of [i.1]	Spectral efficiency b/s/Hz	Modulation Index *h*	Pulse shape AV	Code rate	FEC Conv. Code	Normalized Carrier Spacing (*fT*)	Eb/No(dB)@ PER = 10^{-3}	Eb/No(dB)@ PER = 10^{-5}
9	0,5	2/5	α_{RC} = 0,98	0,5	$(5,7)_{octal}$	2,000	1,75	2,0
10	0,75	1/3	α_{RC} = 0,75	0,5	$(5,7)_{octal}$	1,333	2,3	2,6
11	1,10	2/7	α_{RC} = 0,75	2/3	$(5,7)_{octal}$	1,210	3,2	3,5
12	1,25	2/7	α_{RC} = 0,75	2/3	$(5,7)_{octal}$	1,067	3,6	4,2
13	1,50	1/4	α_{RC} = 0,75	4/5	$(15,17)_{octal}$	1,067	5,4	6,0

Table 10.12: Simulation Results for bursts containing 1 504 information bits

Waveform ID, Table A-2 of [i.1]	Spectral efficiency b/s/Hz	Modulation Index *h*	Pulse shape AV	Code rate	FEC Conv. Code	Normalized Carrier Spacing (*fT*)	Eb/No(dB) @ PER = 10^3	Eb/No(dB)@ PER = 10^{-5}
15	0,5	2/5	α_{RC} = 0,98	0,5	$(5,7)_{octal}$	2,000	1,6	1,8
16	0,75	1/3	α_{RC} = 0,75	0,5	$(5,7)_{octal}$	1,333	2,2	2,4
17	1,10	2/7	α_{RC} = 0,75	2/3	$(5,7)_{octal}$	1,210	3,0	3,4
18	1,25	2/7	α_{RC} = 0,75	2/3	$(5,7)_{octal}$	1,067	3,5	3,9
19	1,50	1/4	α_{RC} = 0,75	4/5	$(15,17)_{octal}$	1,067	5,2	5,6
20	1,80	1/5	α_{RC} = 0,625	6/7	$(15,17)_{octal}$	0,974	8,1	9

Examples of performance curves measured as a function of packet error ratio (PER) are illustrated in figures 10.6 and 10.7.

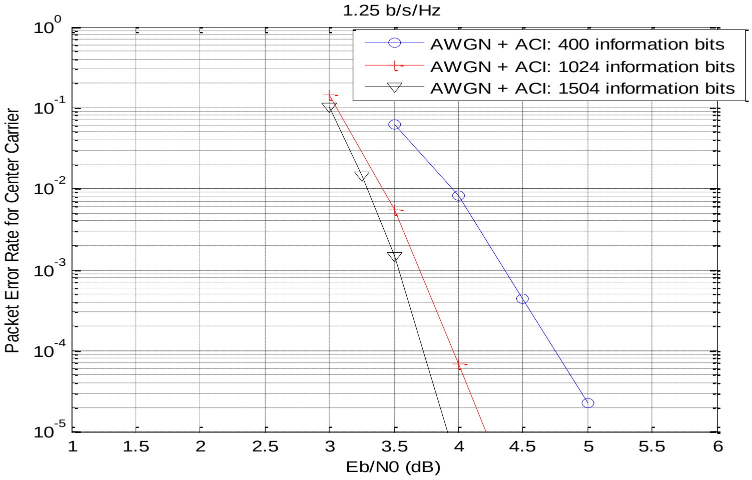


Figure 10.6: Performance results for CC-CPM and spectral efficiency = 1,25 bits/sec/Hz

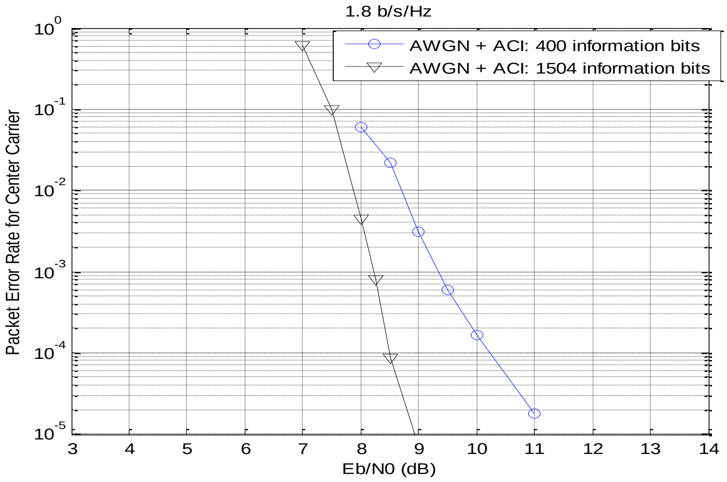


Figure 10.7: Performance results for CC-CPM and spectral efficiency = 1,8 bits/sec/Hz

10.3 RLE Efficiency

The RCS2 return link encapsulation protocol (RLE) has been designed based on a number of assumptions where one of the most critical one is efficiency. In particular, the protocol is designed such that the overhead, in terms of additional bits compared to the payload, be as low as possible for different desired operation scenario. It is worth noting that the relatively small burst sizes in the return link are especially challenging compared to that on the DVB-S2 forward link with a nominally longer frame size.

Based on the additional assumption that RLE runs at the link level and that the link is tightly controlled by the hub or NCC, preference has been given to out-of-band signalling for protocol operation when useful. This means that protocol options that are not expected to change during a single session are not signalled over the air interface because this would cost additional bits which would always have the same value. Nevertheless, several options are supported by the RCS2 LLS that can influence efficiency.

This section will analyse the costs associated with these options and will also make some general statements about the RLE efficiency.

The encapsulation efficiency will be measured as the ratio between the number of useful bits (the bits of the IP packet, for example) and the total number of bits in the payload of the physical layer burst (before spreading and CRC). Ideally, this ratio should be as high as 1, meaning that each bit on the air interface is a useful bit. In practice however, there are a number of sources for bits that are not considered useful in this context, as presented below:

- Bit padding. The RLE protocol is byte-oriented meaning that it can use the burst payload only in multiple of 8 bits. If the burst payload length is not a multiple of 8 bits, the remaining bits cannot be used. By careful design of the coding this does not occur with the standard RCS2 burst sizes.
- **Padding.** Padding can occur at the end of a burst for two reasons:
  - There is no data available to fill the burst with another RLE packet.
  - The available space at the end of the burst is too short to start another RLE packet.
- **Per-burst overhead**. This is overhead that occurs once in a burst. For RLE this consists of:
  - **Signalling byte**. This is the optional first byte of the burst payload which signals the length of the burst label and the length of the frame label. Because in all standard RCS2 RLE profiles these labels have the same length (see Table 7-10 in [i.1]) this byte is completely optional and is used only if the `use_explicit_payload_header_map` field of the Frame Payload Format Descriptor is set to 1 (see section 6.4.17.14 of [i.1]).
  - **Burst label.** For transparent star configurations this label is present only for the random access methods in which case its size is fixed for each method and defined in Table 7-10 of [i.1].

  - **Burst CRC.** No CRC is used in RCS2 because there is already a CRC present for error detection below the spreading sub-layer.

- **Per-fragment overhead**. This is overhead that occurs for each RLE packet (which carry higher layer packets or packet fragments). This overhead consists of the following parts:

  - **RLE packet header**. This is the two byte header which starts each RLE packet.

  - **Fragment label**. This is the label that optionally maybe attached to each RLE packet. For all standard RLE profiles this label has the length 0.

  - **Start packet header**. For RLE packets that contain the start of a higher layer packet (but do not contain the complete higher layer packet) there is an additional two byte header.

  - **Reassembly check**. RLE packets that contain the end of a higher layer packet (but not the start of it) contain the reassembly check bytes. This is either a 1-byte sequence number or a four byte CRC. Whether to use the sequence number or the CRC is signalled by the hub in the allow_alpdu_crc and `allow_alpdu_sequence_number` fields of the Frame Payload Format Descriptor (see section 6.4.17.14 of [i.1]). If both are set the terminal can decide on which mechanism to use. For each fragmented higher layer packet the decision is signalled in-line in the start packet header.

- **Per packet overhead**. This is overhead that occurs once for each higher layer packet. It is produced by the encapsulation sub-layer and consists of the following parts:

  - **Protocol type field**. This field may be 0, 1, 2 or 3 bytes long depending on the actual protocol type, the `allow_ptype_omission`, the `use_compressed_ptype` and the `implicit_protocol_type` fields of the Frame Payload Format Descriptor (see clause 6.4.17.14 of [i.1]).

  - **Packet label**. This field maybe 0, 1 3 or 6 bytes long if the standard RLE profiles are used. For a transparent star only the lengths 0 and 1 apply.

  - **Extension headers**. This is a variable length field the use of which is not specified by the standard.

- In order to analyse the performance of RLE a simulator has been used (see Figure 10.8).

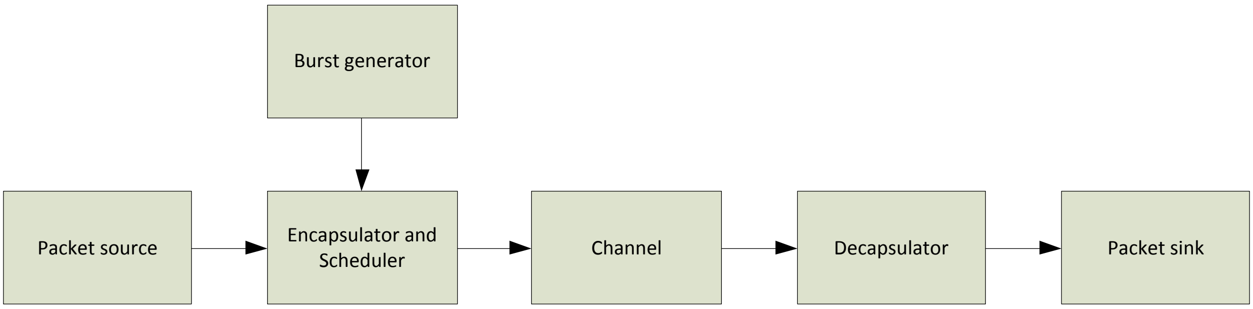


Figure 10.8: RLE performance simulation environment

In this simulator a packet source creates higher layer packets which are sent to the encapsulator and scheduler. The encapsulator and scheduler additionally receive burst descriptions from the burst generator. It creates burst payloads according to these descriptions and forwards them to the channel. The channel may introduce drops and bit errors. The resulting bursts are sent to the decapsulator which extracts the higher layer packets and forwards them to the packet sink.

Two different packet source algorithms have been used for the following analysis:

- A source producing packets of equal size with a fixed rate, a fixed label and a fixed protocol type (CPR source). Simulations have been done for the following packet sizes: 20, 40, 60, 90, 120, 180, 270, 400, 600, 800, 1 000, 1 200, 1 400 and 1 500 byte.

- A source producing multiple streams of packets. Each of the streams can be configured independently and produces packets with a fixed rate, normal distributed packet sizes and fixed label and protocol type (NRM source).

For the burst generator a single algorithm is used:

- A generator producing configurable combinations of ModCod and burst length for fixed amounts of time (LIN generator). The time for a single combination has been set to 600 seconds for all simulations.

Because measurements are required only at the encapsulator, the channel is ideal for all simulations and the decapsulator just drops the packets.

The simulations have been done with fully loaded links so that padding because of no available data does not occur. This kind of padding is related to resource management not to protocol efficiency.

10.3.1 Optimum efficiency case

Optimum performance can be reached with the following configuration:

- No burst signalling byte used.
- User traffic and management traffic carried in a single SVN (single SVN operation).
- Use of sequence number for reassembly check.
- Only IP and signalling traffic.
- Default protocol type is 0x30 (IPv4/IPv6).
- Protocol type compression allowed.
- No extension headers used.

Under these conditions the maximum efficiency of RLE is reached which is shown in Figure 10.9. This figure shows the efficiency for DAMA traffic where no burst label is used.

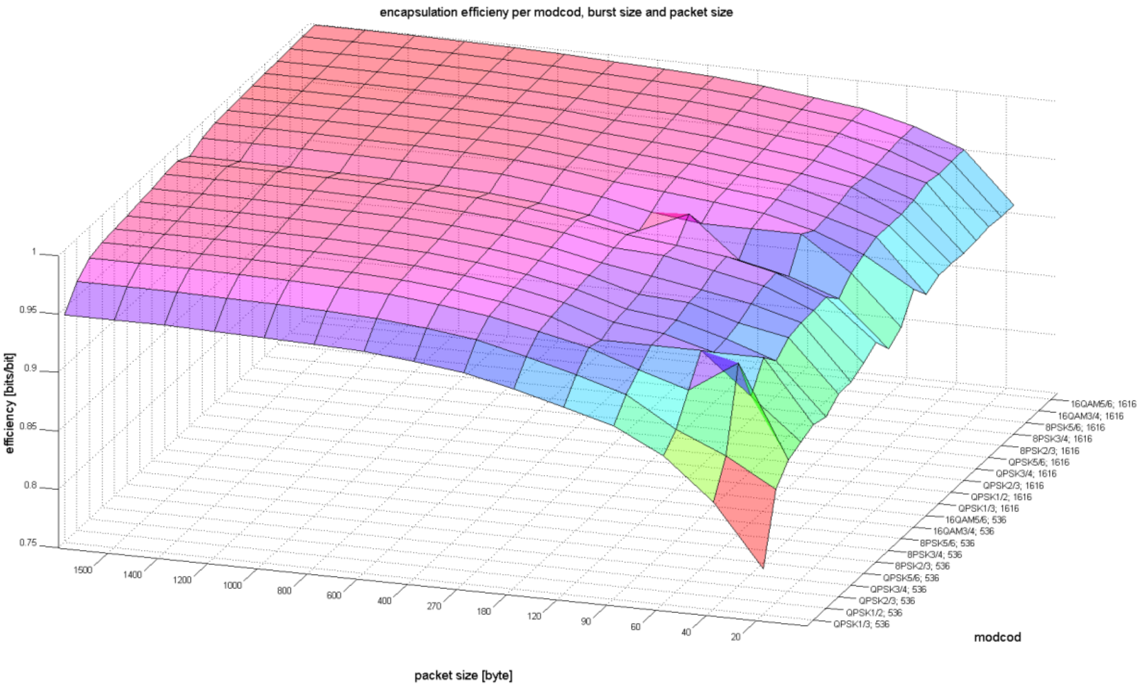


Figure 10.9: RLE optimum performance case

The figure shows the encapsulation efficiency for each combination of the selected set of packet sizes, the standard combinations of the modulation, coding and burst sizes corresponding to two groups of Linear Modulation waveforms (536 and 1 616 symbols) as defined in Table A-1 of the normative document [i.1].

The most critical combination is small packets and small bursts. The lowest efficiency reached is around 0,74 which means the maximum overhead is 26 %. For large bursts and large packets the efficiency reaches 0,994 meaning an overhead of 0,6 %. Small packet sizes (resulting from header-compressed TCP ACKs or VoIP packets) are generally more critical than small burst sizes.

The spikes in the figure are resonance effects when the RLE packet sizes and the burst sizes have a ratio denoted by small numbers.

For the CRDSA encapsulation efficiency is smaller than that for DAMA because of the burst label. Figure 10.10 compares the efficiency of DAMA bursts vs. CRDSA bursts for the lowest Modcod and two burst sizes.

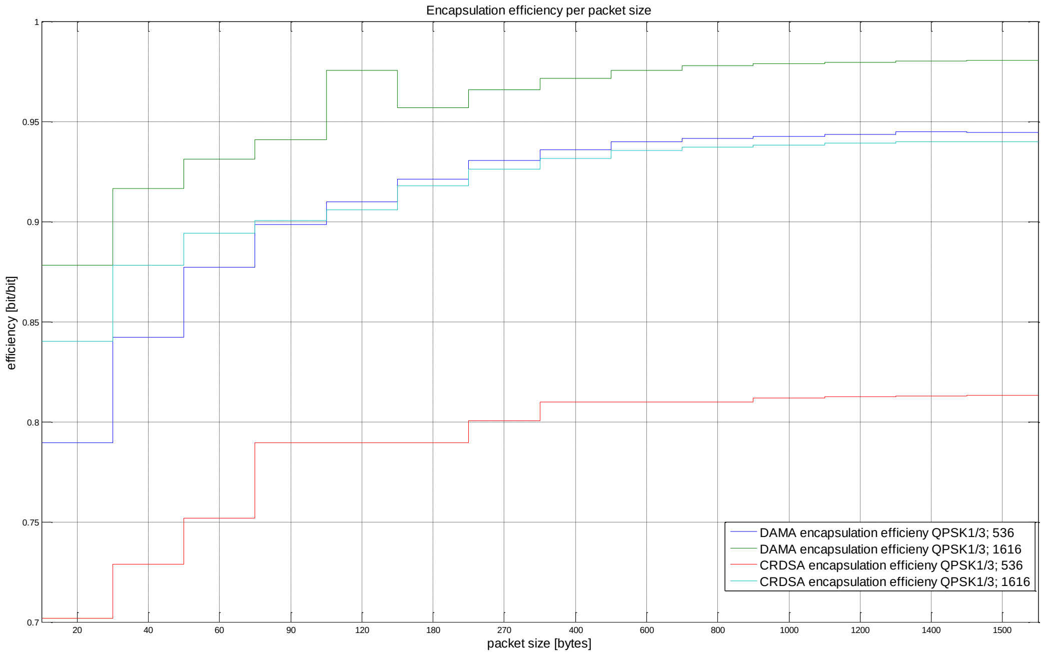


Figure 10.10: DAMA vs. CRDSA encapsulation efficiency

For the minimum bursts the maximum overhead is 30 % (for very small packets) and never goes below 19 % even for very large packets. For the large bursts the numbers are 16 % and 6 % resp. When configuring a system for CRDSA larger bursts should be considered instead of the minimum bursts.

10.3.2 Default protocol and protocol type compression

The protocol type field that normally occupies two bytes for each higher layer packet can be compressed in RLE by two mechanisms: default protocol types and compressed protocol types.

First the hub can provide the terminals with a default protocol type in `the implicit_protocol_type` field of the Frame Payload Format Descriptor and set the `allow_ptype_omission` flag (see section 6.4.17.14 of [i.1]). When the terminal transmits a packet which has this protocol type, it can omit it completely and set the `protocol_type_suppressed` flag in the start packet header thus saving two bytes. This is especially helpful for very short packets like header-compressed VoIP or TCP acknowledgements. The `implicit_protocol_type` defines the default protocol type for label types 0 to 2. The default protocol type for label type 3 is fixed to L2 signalling and cannot be changed.

Second the hub can allow the terminal to compress the protocol type to one byte instead of two bytes by setting the `use_compressed_ptype` field of the Frame Payload Format Descriptor. If this bit is set the terminal looks up the protocol type in the table of compressed protocol types (Table 7-3 in [i.1]). If the protocol type is found the one byte value is inserted into the encapsulated packet. If it is not found, the escape value 0xff is used instead and the real two byte protocol type is inserted after the label. In this case the protocol type is actually expanded instead of compressed. In case a system makes heavy use of a protocol type not contained in Table 7-3 of [i.1], a new compressed protocol type may be used for this system in the user defined area (0x80 to 0xfe).

In order to estimate the savings of using default and compressed protocol types simulations are done with two configurations: without default protocol type, but compressed protocol types and without default and compressed protocol types. The results are compared to the optimum case when both mechanisms are allowed.

Figure 10.11 shows the loss in encapsulation efficiency when not using default protocol types, but only compressed protocol types. In this case the default type has been set to 0x30 (IP) and all packets are IP packets. The change in efficiency is independent from the Modcod, but heavily depends on the packet sizes. It is largest for the small packet sizes where can reach 5 % for packets of 20 bytes.

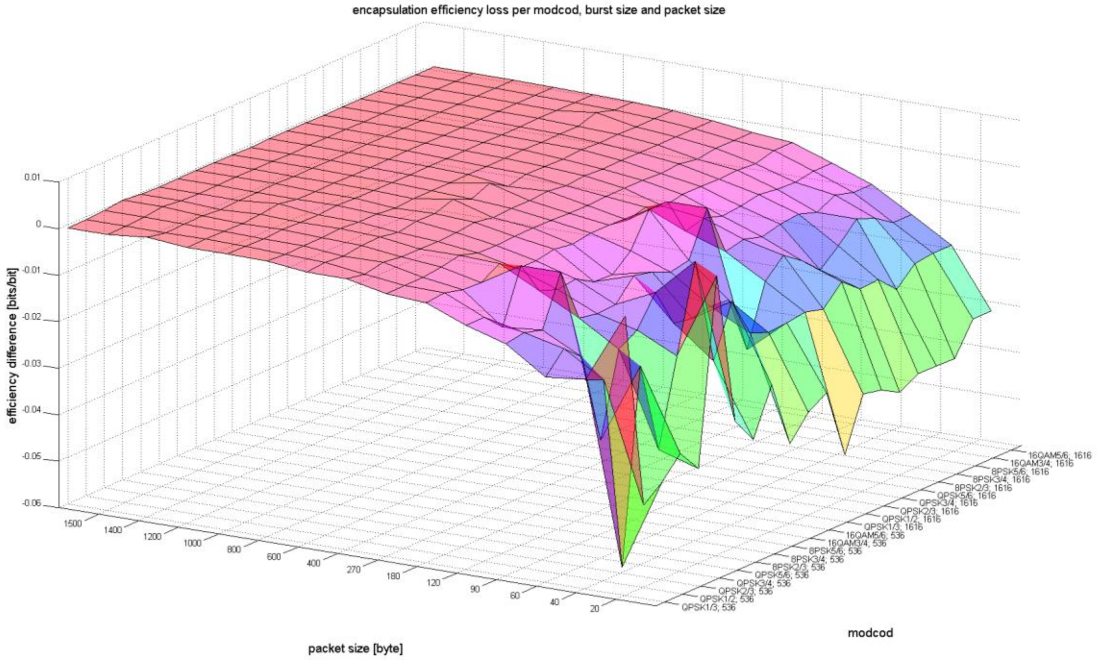


Figure 10.11: Loss of encapsulation efficiency without default protocol type

The case when always the full protocol type is transmitted is show in Figure 10.12. In this case the efficiency maybe up to 8,5 % worse than in the optimum case. The most critical cases are very short higher layer packets.

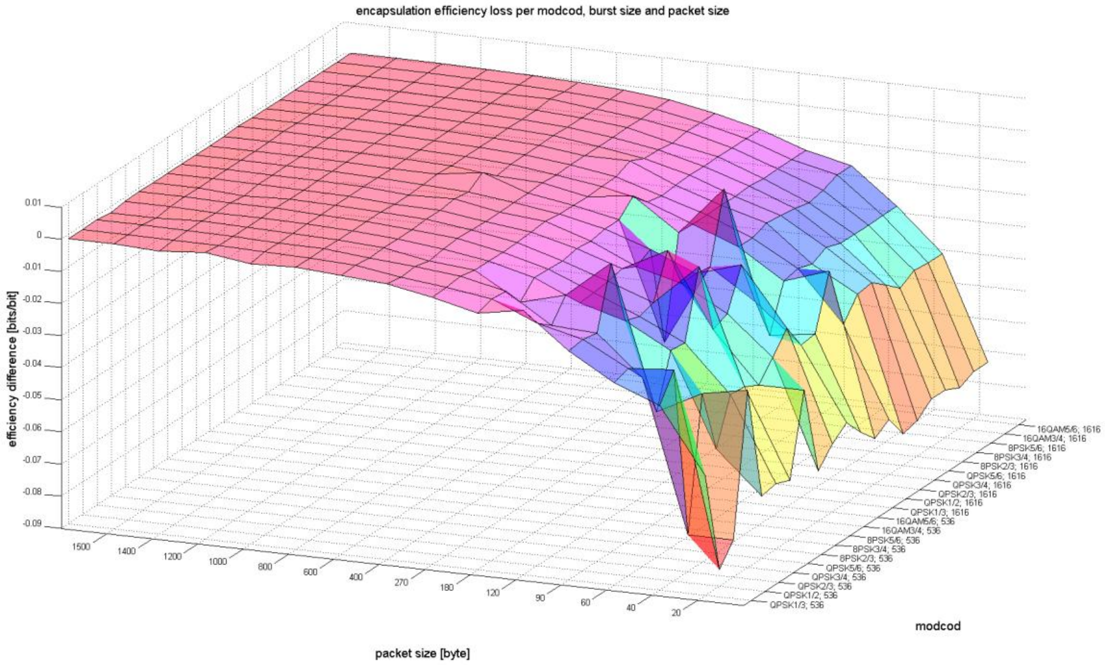


Figure 10.12: Loss of encapsulation efficiency with 2-byte protocol type

For maximum efficiency, especially in systems with a large number of small packets it is strongly recommended to use an appropriate default protocol type and protocol type compression.

10.3.3 Sequence number vs. CRC reassembly control

For error checking of reassembled packets two mechanisms are for seen: one uses a one-byte sequence number and the other a 32-bit CRC. The sequence number is appropriate for normal RCS2 systems with a reasonable packet loss ratio. It works well for packet loss ratios up to 95 %. Use of CRC is recommended for regenerative systems requiring on-the-fly translation of RLE packets to GSE packets. The CRC has been specified to correspond to the CRC of the translated GSE packet. The CRC may also be appropriate for mobile systems with large bursts of packet loss.

The use of the CRC results in a less efficient encapsulation. This effect is shown in Figure 10.13. The effect is most visible for small packet sizes and small to medium burst sizes. In the worst case efficiency drops by approximately 5,3 %.

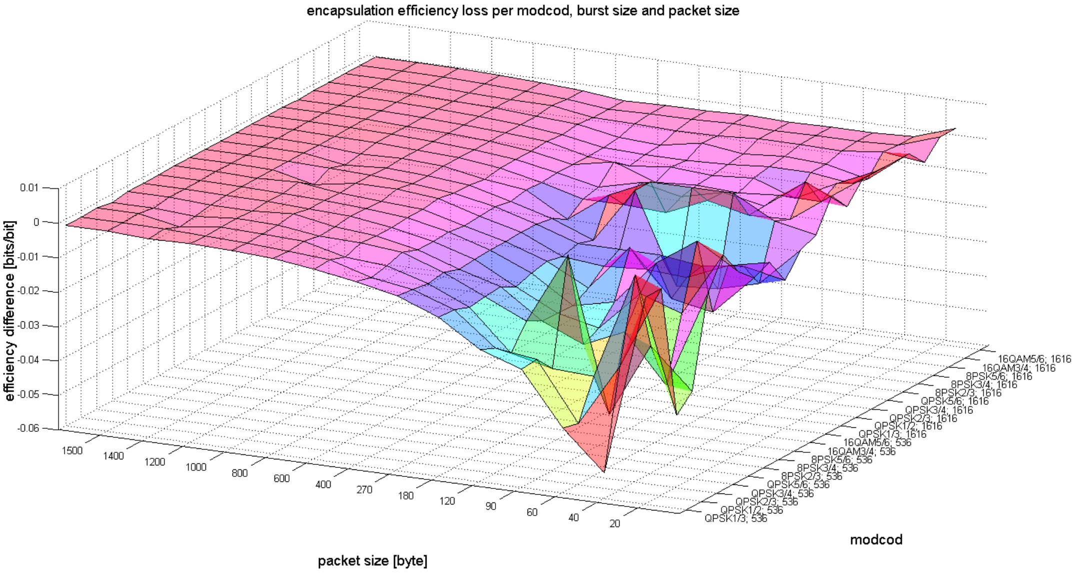


Figure 10.13: Loss of encapsulation efficiency with CRC instead of sequence number

10.3.4 GSE compatibility mode

For on-the-fly translation of RLE into GSE the configuration should not use default or compressed protocol types and use CRC for reassembly check. Additionally fragmentation of the protocol type and the label should be switched off and a packet label will need to be used for destination addressing (assumed to be 3 byte). The resulting efficiency is shown in Figure 10.14. The corresponding difference to the optimal configuration is in Figure 10.15.

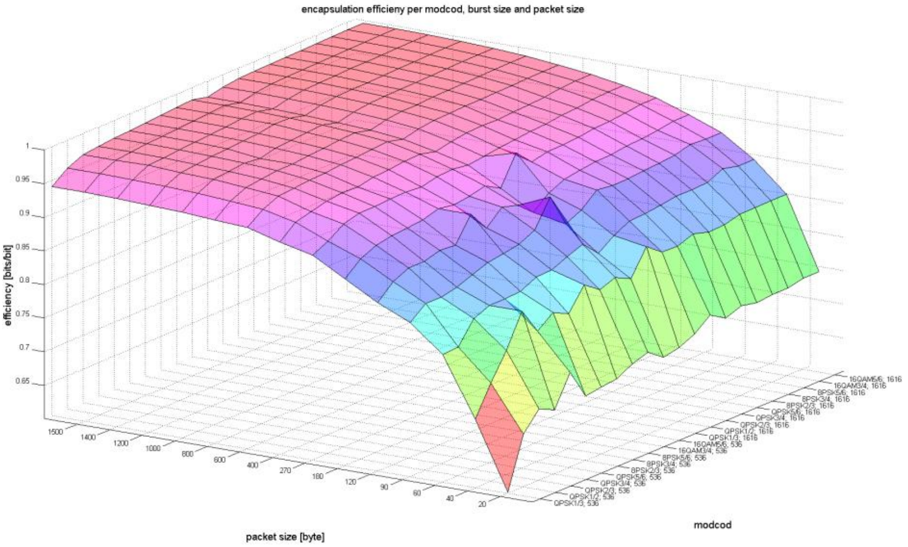


Figure 10.14: Encapsulation efficiency for GSE compatibility mode

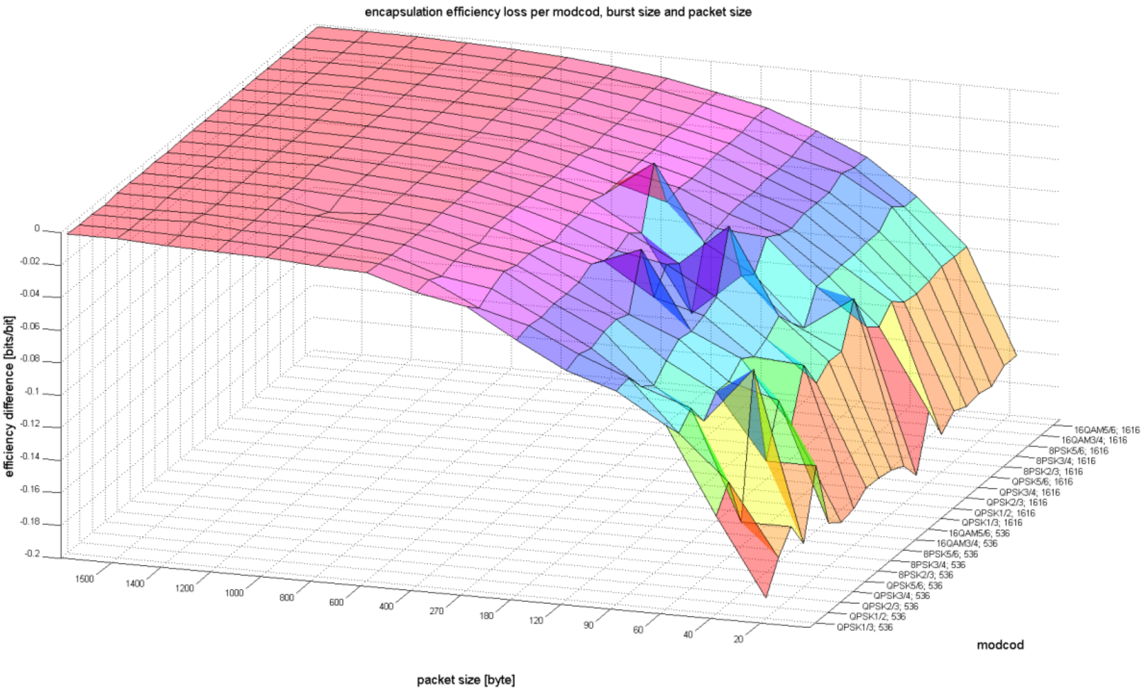


Figure 10.15: Loss of encapsulation efficiency for GSE compatibility mode

If can be see that there is a noticeable loss of efficiency for packets smaller than 500 byte. For very small packets the efficiency reaches just slightly more than 0,6 bit/bit.

The encapsulation efficiency for very small packets and small bursts in this case is just 0,66 bit/bit. For small packets in large bursts (1 608 symbols and 16QAM 5/6) the efficiency is 0,82. For very large packets the figures are 0,91 and 0,99 resp. The difference to the optimum configuration is shown in Figure 10.16. These numbers do not take into account additional fragment or burst labels that may be required for regenerative cases.

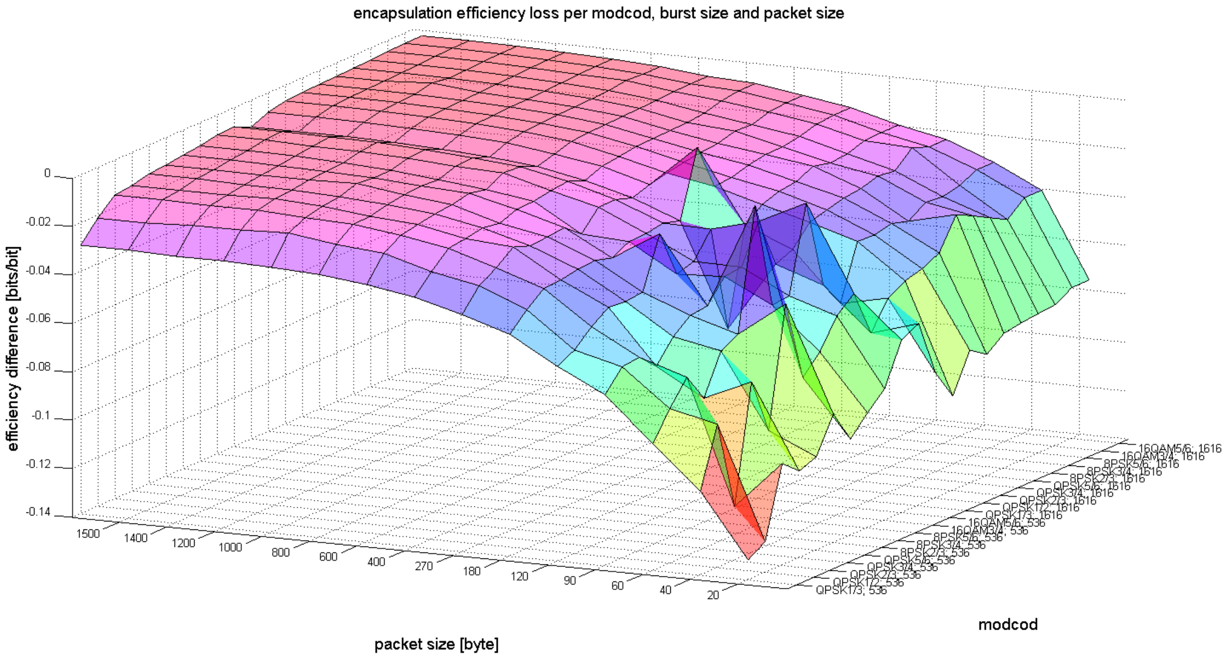


Figure 10.16: Loss of encapsulation efficiency for GSE compatibility mode

10.3.5 Traffic mix

In order to estimate the performance for a mix of packets a simulation with three packet sources and the following parameters has been done:

- Source 1: 400 packets/s with normal distributed packet size (μ = 60 bytes; σ = 40 bytes; minimum size 10 bytes; maximum size 120 bytes). This represents VoIP packets and TCP acknowledgments.
- Source 2: 40 packets/s with normal distributed packet size (μ = 512 bytes; σ = 200 bytes; minimum size 120 bytes; maximum size 1 200 bytes). This represents HTTP requests, DNS packets and other signalling packets.
- Source 3: 40 packets/s with normal distributed packet size (μ = 1400 bytes; σ = 400 bytes; minimum size 1 000 bytes; maximum size 1 500 bytes). This represents TCP data packets.

Two simulations have been carried out: with the optimum RLE configuration and in GSE compatibility mode. The results for the different ModCods and burst sizes are shown in Figure 10.17. For small bursts configuration 1 reaches 0,88 bits/bit while configuration 2 gets 0,82 bits/bit. For large bursts the numbers are 0,98 and 0,99 bits/bit resp.

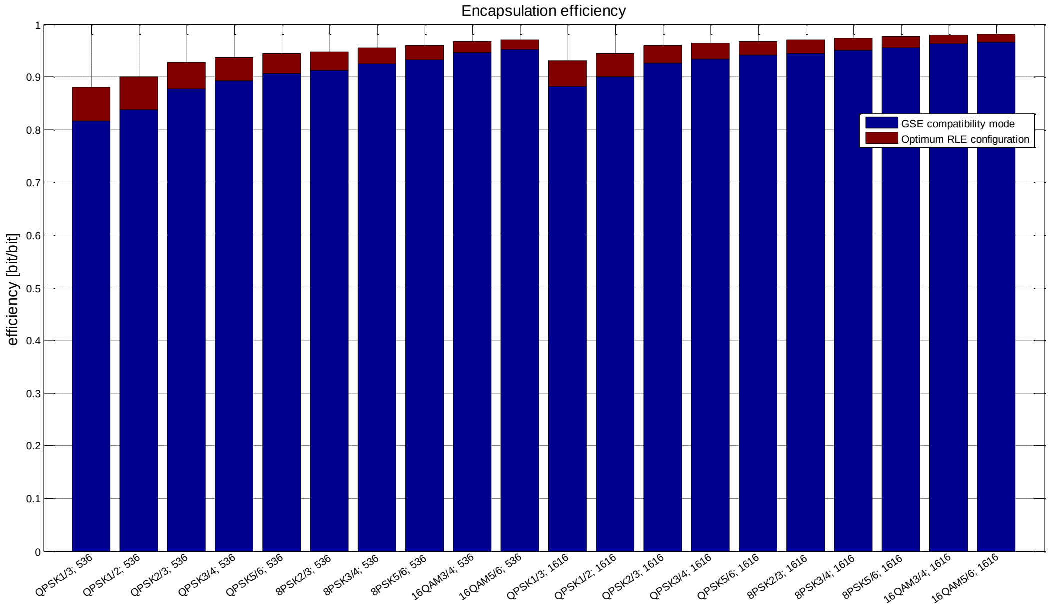


Figure 10.17: Encapsulation efficiency for traffic mix

10.4 Examples of System Performance Evaluation

An example of system application scenario for DVB-RCS2 utilization is described below. Based on a set of system parameters, several examples of link budget analysis as well as system capacity assessment are presented.

10.4.1 System Scenario Definition

As a reference system scenario, a multi-spot beam satellite is considered. The system coverage is Europe using a single satellite located at 33° East using Ka-band frequency on the feeder and user links. The user link coverage consists of 98 beams. The user beams are served by 7 gateways (as shown in Figure 10.19).

The baseline frequency plan is illustrated in Figure 10.18. A single polarization frequency re-use based on 4 colouring scheme has been considered. The frequency plan is based on a 4 colour reuse scheme with 125 MHz per spot beam. The forward and return user links are assigned to different polarizations as shown in Figure 10.18.

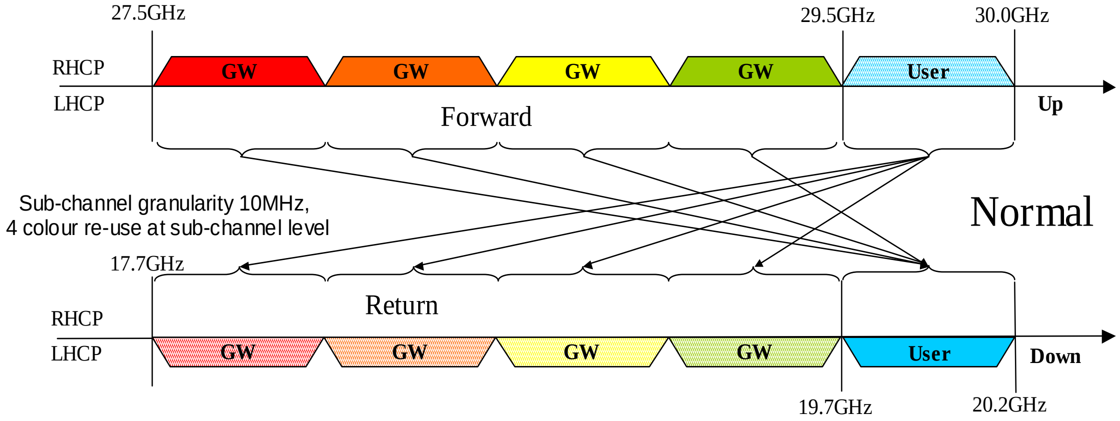


Figure 10.18: Four-Colour, Single Polarization Frequency Plan

The receiver antenna gain (on board or the satellite) is presented in Figure 10.19.

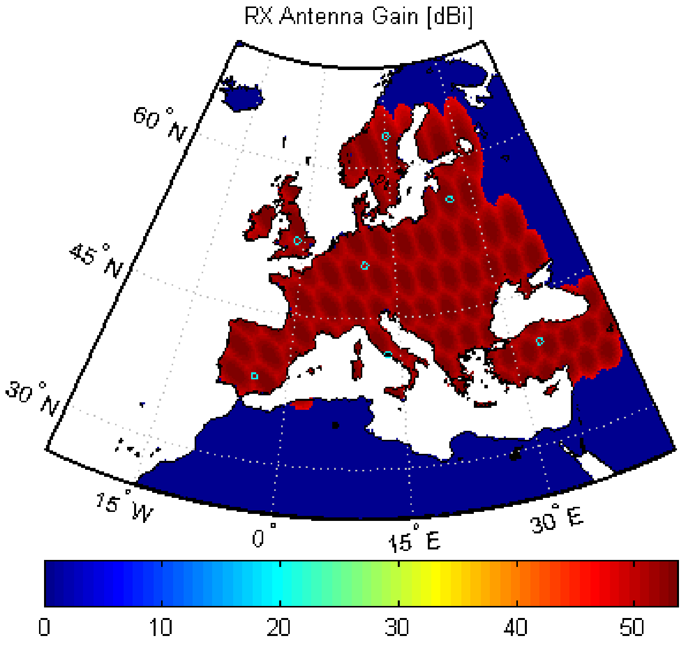


Figure 10.19: On-board Rx antenna gain on the user link

10.4.2 Link Budget Examples

For the system scenario defined in clause 10.5.1, link budget analysis is presented. Three RCSTs located at different geographical locations are considered. Figure 10.20 illustrates RCST 1, 2, 3 which are served by gateways A, B and C respectively.

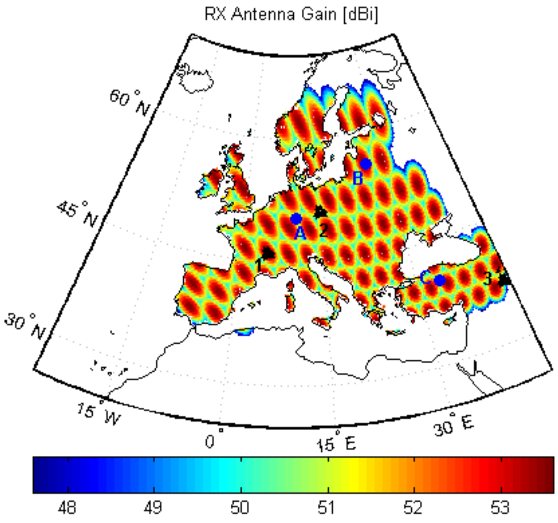


Figure 10.20: On-board Rx antenna gain on the user link

Table 10.13 provides a summary of EIRP realization for RCSTs at three different geographical locations. For these RCSTs a nominal SSPA saturated power of 2 Watts is considered. An additional power variation per RCST has been considered to take into account saturated power variations over the terminal population. The nominal OBO setting is according to the modulation scheme used by the RCST (as noted in the table).

Table 10.13: Examples of EIRP Values

Examples of RCST Tx characteristics			RCST #1	RCST #2	RCST #3
RCST geographical Location	Latitude:	(deg)	46	51	37,25
	Longitude:	(deg)	5	15	43,25
	Altitude:	(m)	292	341	1312
Elevation angle:		(deg)	30,33	29,18	45,48
Up-Link Tx frequency:		(GHz)	29,75	29,75	29,75
Nominal SSPA Saturated Output Power: (see note 1)		(W)	2	2	2
Terminal Power Fluctuation:		(W)	0,05	0,17	0,1
Terminal OBO (see note 2)		(dB)	0,97	0,56	2,29
Terminal TX Losses (see note 3):		(dB)	1,5	1,5	1,5
Terminal Antenna Diameter:		(m)	0,6	0,6	0,6
Terminal Antenna Efficiency:			0,6	0,6	0,6
Antenna Peak Gain:		(dBi)	43,22	43,22	43,22
EIRP:		(dBW)	43,87	44,52	42,64
NOTE 1: The maximum output power is assumed to fluctuate from unit to unit and in time with $3\sigma \leq 0,5$ dB. NOTE 2: The OBO setting is a function of the modulation: 0,5 dB for QPSK, 0,7 dB for 8PSK and 2,5 dB for 16QAM. NOTE 3: Tx Losses include both coupling losses (typically 0,5 dB) as well pointing losses (considered below 1 dB).					

A clear sky link budget has been evaluated at three different geographical locations taking into account system and payload parameters. Results are presented for three different baud rates corresponding also to three different selection of modulation and coding per each RCST.

Table 10.14: User Up-Link

DVB-RCS2 Link Budget for Linear Modulation Schemes		**RCST#1**	**RCST#1**	**RCST#1**
RCST Tx (See note 1)				
Carrier Symbol Rate:	(Mbaud)	7,69	3,85	1,93
Modulation		8-PSK	8-PSK	16QAM
Code Rate		5/6	3/4	5/6
Occupied Bandwidt (20 % Roll-off)	*(MHz)*	*9,22*	*4,62*	*2,32*
Up-Link Tx frequency:	(GHz)	29,75	29,75	29,75
EIRP per carrier (see note 1)	(dBW)	43,87	44,52	42,64
Atmospheric Attenuation (w/o rain):	(dB)	0,53	0,5	0,3
Polarization Losses:	(dB)	0	0	0
Rain Attenuation	(dB)	0	0	0
Slant Range:	(Km)	38 580,39	38 683,54	37 377,87
Path Loss:	(dB)	213,64	213,66	213,36
SAT RX				
Sat RX Gain:	(dBi)	53,56	50,13	47,92
Sat RX NF:	(dB)	2	2	2
Sat RX Total T:	(K)	490,94	490,94	490,94
Sat RX G/T:	(dB/K)	26,65	23,21	21,01
Up-link C:	(dBW)	-116,74	-119,52	-123,1
Up-link C/N0:	(dBHz)	84,95	82,17	78,59
Up-link C/N:	(dB)	16,1	16,31	15,74
Up-link C/I co-channel and adj-channel (see note 2)	(dB)	14,72	12,52	18,65
Up-link C/IM (see note 3)	(dB)	22,89	21,22	22,71

NOTE 1: RCST locations, EIRP values and other characteristics are outlined in Table 10.13.

NOTE 2: C/I is computed both for co-channel interference from co-colour beams as well adjacent channel interference taking into account the OBO of the adjacent carriers. Carrier spacing of 1,2 R_s is assumed.

NOTE 3: C/IM reflects the impact of the non-linear distortion taking into account the spectral regrowth and out-of-band power computed per each modulation and coding.

Table 10.15: Feeder Down-link

DVB-RCS2 Return Link budget		RCST#1	RCST#2	RCST#3
SAT TX				
Down-Link frequency:	(GHz)	18,7	18,7	18,7
Satellite Saturated Output Power:	(W)	130	130	130
Sat TX OBO:	(dB)	4	4	4
Satellite resulting fixed Gain:	(dB)	108,87	108,88	108,87
Sat TX Losses:	(dB)	1,55	1,55	1,55
Sat TX Gain:	(dBi)	50,32	53,25	51,63
EIRP per carrier:	(dBW)	40,9	41,06	35,85
Atmospheric Attenuation (w/o rain):	(dB)	0,34	0,38	0,21
Down link polarization Losses:	(dB)	0,2	0,2	0,2
Path Loss:	(dB)	209,61	209,63	209,33
GATEWAY RX				
GW Latitude:	(deg)	50,33	56	40,04
GW Longitude:	(deg)	10,17	25,93	33,77
GW Elevation:	(deg)	28,5	25,9	43,72
GW Altitude:	(m)	333	140	943
GW Slant Range:	(Km)	38 744,81	38 984,51	37 502,18
GW Antenna Diameter:	(m)	8	8	8
GW Antenna Efficiency:		0,67	0,67	0,67
GW RX Gain:	(dB)	62,17	62,17	62,17
GW RX Noise Figure:	(dB)	2	2	2
GW RX Antenna Temperature:	(K)	44,01	52,15	31,31
GW RX Total T:	(K)	314,71	322,85	302,01
GW RX G/T:	(dB/K)	37,19	37,08	37,37
GW RX Losses:	(dB)	1	1	1
RX Signal Power C:	(dBW)	-108,08	-107,98	-112,72
Down Link C/N0:	(dBHz)	95,51	95,46	91,05
Down Link C/N:	(dB)	26,65	29,6	28,2
Down Link C/I:	(dB)	33,49	27,09	43,18
Total C/(N+I):	(dB)	11,8	10,46	13,25
Required threshold per Selected Modulation and Coding (See note)	(dB)	9,3	11,00	13,00
NOTE: Reported threshold values correspond to PER = 10^{-5} for long bursts (1 616 symbols) Compared to AWGN threshold results reported in Table 10.6, implementation losses and performance degradation due to phase noise has also considered.				

10.4.3 System Capacity Evaluation

Examples of system capacity evaluation for the return channel are presented. The baseline system is as per the system definition presented in clause 10.4.1 corresponding to a multi-beam satellite system. The focus of this analysis is on the return link using DVB-RCS2 waveforms according to Linear Modulation.

10.4.3.1 System Assumptions

In order to perform system analysis certain system assumptions have been considered:

a) within each beam bandwidth, carrier segregation is applied (i.e. co-channel carriers have the same bandwidth);

b) the organization of carriers within a beam is assumed to be repeated for all the user beams;

c) the network is fully loaded and a perfect packet scheduler is considered, implying that all the bursts are filled;

d) both co-channel and adjacent channel interferers are always considered in clear sky, which is a worst case assumption although quite pertinent for both availability and capacity computations;

e) uniform traffic request is assumed;

f) no external interference (e.g. inter satellite interference) was considered in the simulation;

g) no adaptive uplink power control for satellite terminals is considered.

Performance analyses have been carried out for two different classes of RCST terminals (non-coexisting in the same network), one with 2 W nominal saturation power and one with 0,5 W saturation power. System performance results are reported for both classes.

System performance evaluation has taken into account a minimum peak data rate requirement of 10 Mbits/sec over 90 % of the coverage area. This requirement has been considered for both set of simulations based on 0,5 W and 2 W nominal saturation power.

It should be noted that the reported results here represent merely an indication of the overall system performance and are meant to serve as examples. The selection of the system parameters, waveforms and system assumptions will have significant impact on the actual performance results.

10.4.3.2 Selected waveforms for system simulations

Table 10.16 shows a set of selected waveforms used for system performance assessment The required thresholds values for each waveform includes RX implementation losses based on a bench system implementation prototype.

Table 10.16: Selected Linear Modulation Waveforms and their Performance

Waveform ID Table A-1 of [i.1]	Burst Length	Modulation	Code rate	E_s/N_0 (dB) @ PER = 10^{-5} AWGN Ideal Synch.	Required E_s/N_0 (dB) @ PER = 10^{-5} Including RX Impl. Loss (See note)	Bits/Symbol
13	1 616	QPSK	1/3	-0,51	0,0	0,61
14	1 616	QPSK	1/2	1,71	2,3	0,93
15	1 616	QPSK	2/3	3,69	3,9	1,30
16	1 616	QPSK	3/4	4,73	5,0	1,47
17	1 616	QPSK	5/6	5,94	6,1	1,64
18	1 616	8PSK	2/3	7,49	8,2	1,75
19	1 616	8PSK	3/4	8,77	9,3	1,98
20	1 616	8PSK	5/6	10,23	11,0	2,19
21	1 616	16QAM	3/4	10,72	11,6	2,66
22	1 616	16QAM	5/6	12,04	13,0	2,96
NOTE: Thresholds include the receiver implementation loss and performance degradation due to realistic receiver algorithms.						

10.4.3.3 Interference and Fading Mitigation Techniques

The selected DVB-RCS2 waveforms, as listed in Table 10.16, provide a range of operating points (in terms of SNIR requirements) while maintaining a quasi-constant burst sizes. From a system perspective, this allows a flexible use of link adaptation techniques such as adaptive coding and modulation (ACM) or dynamic symbol rate adaptation (DRA) as well as combination of both.

By changing the physical layer configuration based on the channel conditions, ACM optimizes the system efficiency and consequently its throughput. The use of low symbol rate carriers (in addition to ACM) can enhance the system availability by switching the satellite terminals that experience high fading onto a low symbol rate carrier. In such conditions, the link is closed by increasing the instantaneous SNIR due to the increased signal power spectral density, although with a penalized data rate.

However, the use of DRA is not only to maintain the link availability in the presence of atmospheric fading, but also to cope with the variable co-channel and adjacent channel interference in multi-beam satellite systems using MF-TDMA as return link access scheme. In particular, the use of combined DRA and ACM can considerably reduce the effect of adjacent channel interference while maintaining a higher peak rate distribution. The use of combined DRA and ACM techniques in DVB-RCS2 systems has been recently investigated and reported in [i.45]. Similar methodology has been used here for system capacity analysis.

10.4.3.4 System Performance Results

Systems performance results are summarized in Table 10.17 and Figures 10.21 to 10.23. Results are obtained using a computer system simulation tool taking into account the system assumptions described in previous sections. Further details regarding the simulation tool specifications and system models can be found in [i.45].

Comparative system performance analyses indicate that the use of combined ACM and DRA will allow a higher distribution of peak data rate among the satellite terminals when compared to a system using ACM only, without compromising other performance measures such as the averaged total offered capacity and service availability, as summarized in Table 10.17.

In this context, the availability figures indicate percentage of coverage area where the return link is available at least 99,7 % of the time.

As indicated in Table 10.17 for systems scenario with 2 W transmit power per ST, there is even some gains in terms of the average capacity and service availability when combined DRA and ACM is deployed.

Figure 10.22 and Figure 10.23 illustrates the utilization of 10 Linear modulation waveforms (shown in terms of modulation and coding distribution) for two different adaptive return link techniques (namely (1) ACM and (2) combined ACM and DRA). The results correspond to the system with 0,5 W and 2 W transmitting power RCSTs respectively.

Table 10.17: Capacity Evaluation Results Linear Modulation, two values of TX power

RCST Class	Tx Power = 0,5 W		Tx Power = 2,0 W	
Link Adaptation Technique	ACM	ACM+DRA	ACM	ACM+DRA
Capacity (Gbits/sec)	16,1	15,6	20,6	21,2
Spatial Availability (see note 1)	98,32 %	98,54 %	98,4 %	100 %
average spectral efficiency (bit/sec/Hz) (see note 2)	1,32	1,28	1,69	1,74
NOTE 1: The term Availability refers to Spatial availability of service for 99,7 % of the time availability. NOTE 2: The term "average" refers to an averaging both in space, i.e. over the coverage, and time, i.e. over the various fading events with the related probabilities.				

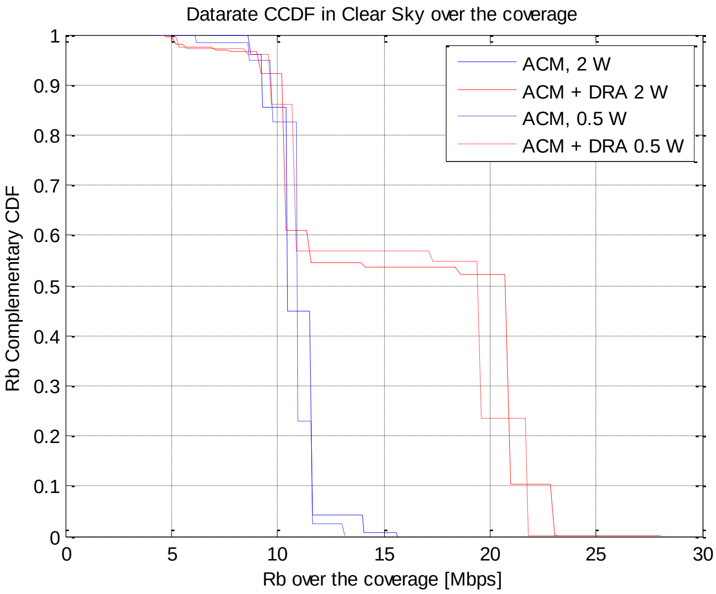


Figure 10.21: Complementary Cumulative Distribution of available bit rate in Clear Sky over coverage area

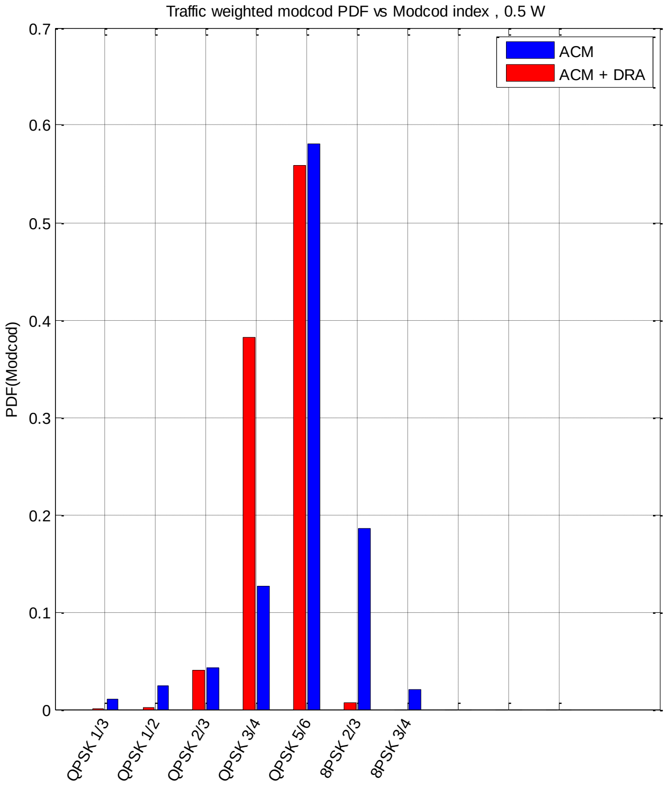


Figure 10.22: The utilization of different modulation/coding schemes for ACM and combine ACM and DRA schemes with a nominal Tx power of 0,5 W

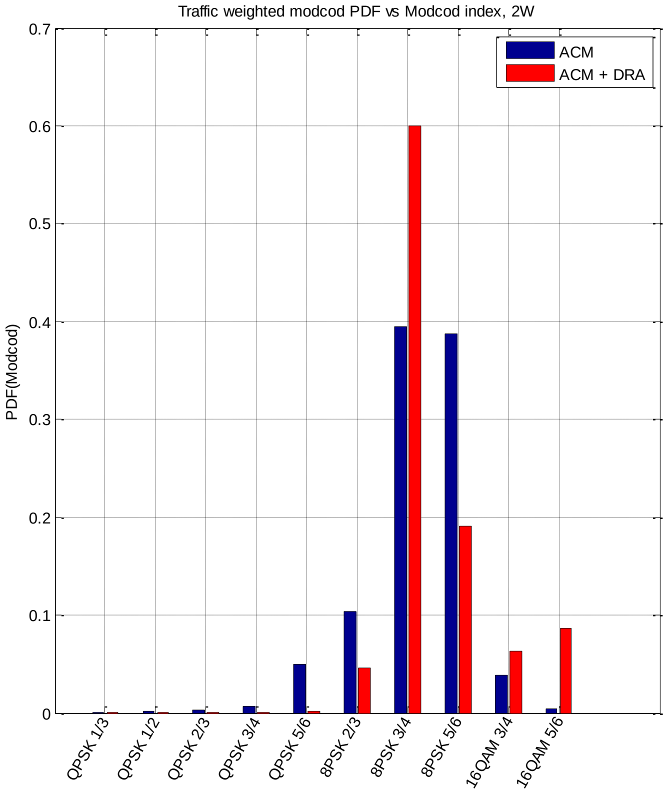


Figure 10.23: The utilization of different modulation/coding schemes for ACM and combine ACM and DRA schemes with a nominal Tx power of 2 W

Annex A: Generalized CPM Waveform Definition

A.1 Introduction

The normative document [i.1] defines a combination of convolutional coding and continuous phase modulation (CPM) as one of the two physical layer schemes adopted for use in DVB-RCS2 (see clause 7.1 of [i.1]). There are more general approaches to define CPM waveforms for the return satellite interactive channel. Such generalized approach provides a framework that can incorporate future improvements by introducing new features in terms of performance, spectral efficiency and implementation. This clause provides a unified approach in defining CPM waveforms including the CC-CPM normative scheme of [i.1] as well as alternative combinations of coding, interleaver and CPM modulation.

The definition of Generalized CPM waveform has impact on several elements of the normative document for potential future use (as a user defined waveform). Figure A.1 illustrates the general framework for defining generalized CPM waveforms.

Figure A.2 shows two examples to use the general framework to define the normative CPM waveform (CC-CPM) and an alternative waveform using a different FEC, interleaver and modulation parameters.

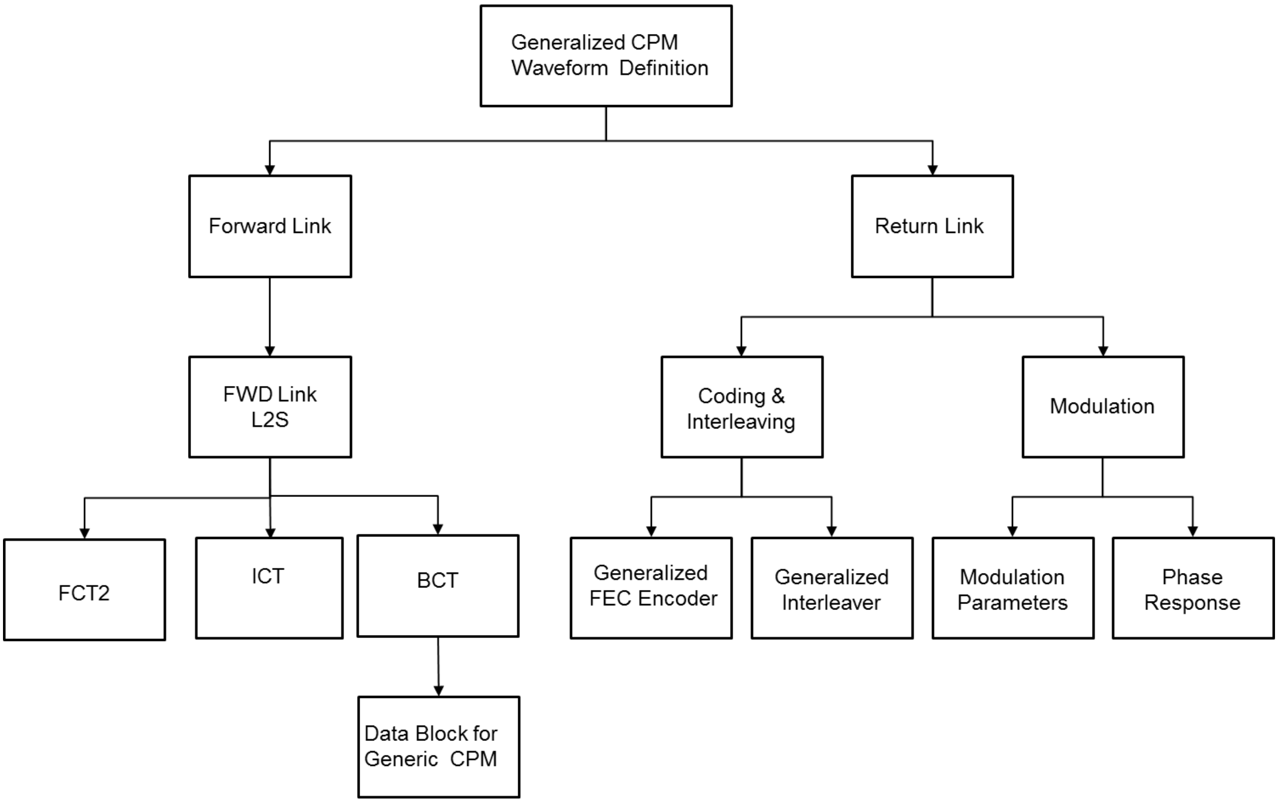


Figure A.1: Generalized CPM Waveform Definition

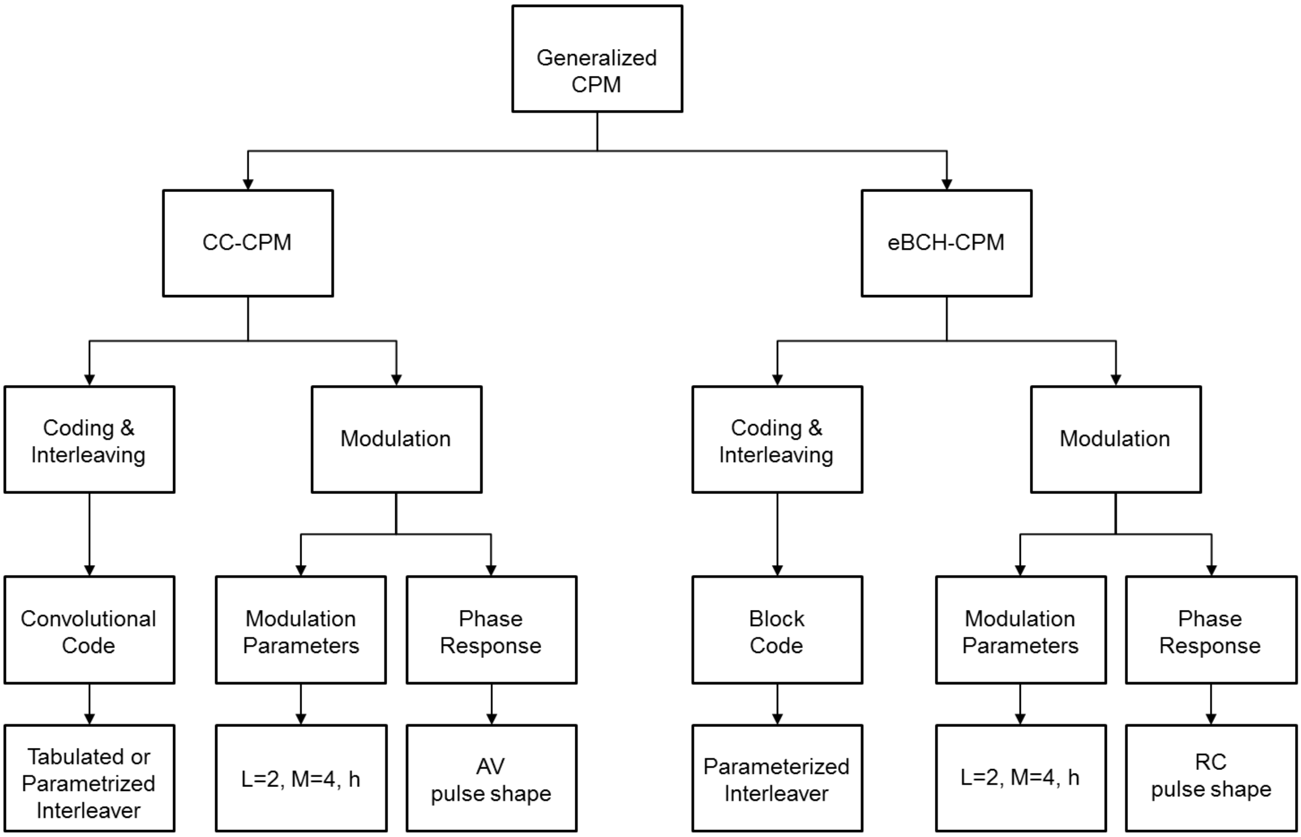


Figure A.2: CC-CPM and eBCH-CPM schemes in Generic CPM mode

A.1.1 CC-CPM scheme

A Concatenated Coding with CPM (CC-CPM) modulation scheme is defined in clause 7.3.5.2 of the normative document [i.1]. The normative document also defined binary, non-systematic, non-recursive convolutional codes as part of the CC-CPM scheme. The constraint length *K* is either 3 or 4.

The generator polynomials for the rate 1/2 constraint length *K*=3 code are:

- $G_{NS1} = 1 + x^2$ (5 in octal)
- $G_{NS2} = 1 + x + x^2$ (7 in octal)

The generator polynomials for the rate 1/2 constraint length 4 code are:

- $G_{NS1} = 1 + x + x^3$ (15 in octal)
- $G_{NS2} = 1 + x + x^2 + x^3$ (17 in octal)

Code rates >1/2 are obtained by puncturing the rate 1/2 code. The puncturing patterns are given in the normative document [i.1].

A.1.2 eBCH-CPM scheme

The functional blocks in the eBCH-CPM encoder include the eBCH encoder, bit interleaver, bit-to-symbol mapping, and CPM modulator. All possible scheme parameters are reported in clause A.4.

Two e-BCH codes are specified: (*k*: information bits of sub-block, *n*: coded bits of sub-block, $d_{\min}$: minimum distance)

- (k=51, n=64, $d_{\min}$=6) adopted for low spectral efficiencies ($\eta \leq 1$);
- (k=113, n=128, $d_{\min}$=6) adopted for high spectral efficiencies ($\eta > 1$).

where *k*, *n* and $d_{\min}$ represent the information bits of sub-block, the coded bits of sub-block, and the minimum distance respectively.

The bit interleaver is based on the following permutation rule:

$$j = (i \times P_1 + P_2) \bmod N \qquad i = 0,1,2,\ldots,N-1$$

where P_1 and P_2 are two prime numbers and *N* is the interleaver length, i.e. the codeword length.

The interleaver is a classical *S*-random interleaver. The algorithm needed to generate the interleaver pattern is summarized in the following steps:

1) Random permutations are created by generating random integers *i*, $1 \le i \le N$ 1 ≤ i ≤ N, without replacement.

2) The permutation rule is defined as follows:

   - each randomly selected integer is compared to a spread value *S* previously selected integers;
   - if the current selection is equal to any *S* previous selections within a distance of ± *S*, then the current selection is rejected;
   - this process is repeated until all *N* integers are selected.

The CPM modulation parameters are specified in clause A.3.2.

A.2 Forward Link L2S

This clause describes necessary changes to Lower Layer Signaling in order to support the Generic CPM scheme.

A.2.1 Description of FL L2S Components

A.2.1.1 The FCT2 content

Table 6-15 of the normative document [i.1] defines all transmission types used in the frame type. In order to support the generalized CPM waveform, a new entry has been added to this table as shown in Table A.1:

- `tx_format_class`: This field indicates the transmission format class of all transmission types used in the frame type. The values are assigned in Table A.1.

Table A.1: Coding of Transmission Format Classes

Value	tx_format_class
0	Reserved
1	Linear Modulation Burst Transmission
2	Continuous Phase Modulation Burst Transmission
3	Continuous Transmission
4	Spread-Spectrum Linear Modulation Burst Transmission
5 to 127	Reserved
128	Generic Continuous Phase Modulation Burst Transmission
129 to 255	User defined

A.2.2 Syntax and Coding of FL Signals for L2S

A.2.2.1 The ICT content

Table 6-16 of the normative document [i.1] specifies the different transmission types. In order to support a generalized CPM waveform, an Interleaver configuration table ICT is added as shown in Table A.2. The ICT Table_id is set to 0xC0 (see Table 6-1 of [i.1]).

Table A.2: Syntax of the Interleaver Configuration Table Content

Syntax	No. of bits		Mnemonic
	Reserved (see note 1)	Information	
interleaver_configuration_table_content() {			
Interleaver_loop_count		8	uimsbf
for (j=1;j<=interleaver_loop_count;j++) {			
parameterized_interleaver	7	1	uimsbf
if(parameterized_interleaver==1){			
interleaver_seed_P1	4	12	uimsbf
interleaver_seed_P2	4	12	uimsbf
spread_value	4	12	uimsbf
}			
else{			
interleaver_id		8	uimsbf
pi_data_size		16	uimsbf
for (k=0;k<pi_data_size;k++) {			
pi_data_bit		1	uimsbf
}			
while (!bytealigned) {			
stuffing_bit		1	uimsbf
}			
}			
}			
NOTE 1: Reserved bits are of type bslbf, and should precede the Information bits on the same line. NOTE 2: The interleaver loop may contain zero, one or more of all the interleavers.			

- interleaver_loop_count: The amount of interleavers present in the next loop.
- parameterized_interleaver: This is a 1 bit field. When set to 1, it stipulates that the CPM bit interleaver permutations be computed using the parameters interleaver_seed_P1, interleaver_seed_P2, spread_value. When set to 0, the interleaver is specified by its interleaver id.
- interleaver_seed_P1: This 12 bit field is an positive prime integer number used in generation the interleaver permutations.
- interleaver_seed_P2: This 12 bit field is an positive prime integer number used in generation the interleaver permutations.
- spread_value: This 12 bit field is an positive integer number used in generation the interleaver permutations.
- interleaver_id: This 8 bit field identifies the interleaver that is defined here.
- pi_data_size: This 16 bit field specifies the data size in bits for interleaver PI, the value should always be a multiple of 12.
- pi_data_bit: This field contains a data bit for the interleaver PI.
- stuffing_bit: Since the UW description, interleaver PI are byte aligned, stuffing bits are present until the next byte boundary. The stuffing bits may take any value and should be discarded by the terminal.

A.2.2.1.1 Data Block for Generic CPM

The data block format providing the configuration for the generic CPM mode is specified in Table A.3.

Table A.3: Data Block format for the Generic CPM Transmission

Syntax	No. of bits		Mnemonic
	Reserved (see note)	Information	
gen_cpm_data_block {			
tx_block_size		8	uimsbf
threshold_es_n0		8	uimsbf
tx_start_offset	12	20	uismbf
modulation_m_h	1	3	uimsbf
modulation_p_h	1	3	uimsbf
encoder_type	1	1	uimsbf
cpm_encoder_memory_length		2	uimsbf
modulation_type		4	uimsbf
for (i=0;i<cpm_encoder_memory_length*16;i++) {			
phase_pulse_sample		16	uimsbf
}			
for (i=0;i<modulation_p_h;i++) {		8	uimsbf
normalization_sequence			
}			
number_of_subblocks		8	uimsbf
subblock_parity		1	uimsbf
encoding_polynomial1		23	uimsbf
encoding_polynomial2		8	uimsbf
subblock_input_length		16	uimsbf
subblock_output_length		16	uimsbf
one_bit_longer_subblocks		8	uimsbf
interleaver_id		8	uimsbf
uw_length		16	uimsbf
for (i=0;i<uw_length;i++) {			
uw_bit;		1	uimsbf
}			
while (!bytealigned) {			
stuffing_bit		1	uimsbf
}			
normalisation_sequence_length		3	
nbr_uw_sequences	1	4	
for (i=0;i<nbr_uw_sequences;i++) {			
uw_sequence_start;		15	uimsbf
normalization_sequence_flag		1	uimsbf
uw_segment_length		8	uimsbf
}			
}			
NOTE: Reserved bits are of type bslbf, and should precede the Information bits on the same line.			

Semantics for the gen_cpm_data_block:

- tx_block_size: The number of consecutive BTUs required for transmission of the physical layer block used by the specific tx_type. This indicates the size of the timeslot required for the burst.
- threshold_es_n0: This is the nominal sensitivity for the transmission type encoded as (5 * threshold) + 120 with the threshold given in dB, and serves as a reference for ACI control as specified in clause 7.3.8 of normative text.
- tx_start_offset: A 20-bit field that gives the nominal offset for burst start from the start of the timeslot in units of NCR ticks.
- modulation_m_h: This 3 bit field specifies the numerator in a fraction representing the modulation index. The numerator m_h equals the value of this field +1.
- modulation_p_h: This 3 bit field specifies the denominator in a fraction representing the modulation index. The denominator p equals the value of this field +1.
- encoder_type: This 1 bit field specifies the encoder type configuration. A value '0' indicates FEC variant 1 (systematic). A value '1' indicates FEC variant 2 (non-systematic).

- modulation_type: This 4 bit field specifies the modulation type and the symbol mapping option, as defined in Table A.4.

Table A.4: Modulation_type value

Modulation_type value	Modulation	Symbol mapping
000	binary	Linear
001	quaternary	Gray mapping
010	quaternary	Linear mapping
011	octal	Gray mapping
100	octal	Linear mapping
101	reserved	Reserved
110	reserved	Reserved
111	user-defined	User-defined

Table A.5: Bit to Symbol mapping for binary linear mapping

MSB LSB b0	Symbol value
0	-1
1	1

Table A.6: Bit to Symbol mapping for quaternary linear mapping

MSB b1	LSB b0	Symbol value
0	0	-3
0	1	-1
1	0	1
1	1	3

Table A.7: Bit to Symbol mapping for quaternary gray mapping

MSB b1	LSB b0	Symbol value
0	0	-3
0	1	-1
1	1	1
1	0	3

Table A.8: Bit to Symbol mapping for octal linear mapping

MSB b2	b1	LSB b0	Symbol value
0	0	0	-7
0	0	1	-5
0	1	0	-3
0	1	1	-1
1	0	0	1
1	0	1	3
1	1	0	5
1	1	1	7

Table A.9: Bit to Symbol mapping for octal gray mapping

MSB b2	b1	LSB b0	Symbol value
0	0	0	-7
0	0	1	-5
0	1	1	-3
0	1	0	-1
1	1	0	1
1	1	1	3
1	0	1	5
1	0	0	7

- cpm_encoder_memory length: This 2 bits specifies the length of CPM encoder.

Table A.10: CPM encoder memory length

MSB b1	LSB b0	Memory length(L)
0	0	1
0	1	2
1	0	3
1	1	Reserved

- phase_pulse_sample: This 16 bit field specifies a sample of the applicable phase pulse.

- number_of_subblocks: This 8 bit field specifies the number of subblocks in the encoder.

- subblock_parity: This 1 bit when set (value = 1) means that there is an additional parity bit for the systematic code. When not set (value = 0) there is no parity bit.

- encoding_polynomial1: This 23 bit field specifies encoding polynomial 1. See clause A.3.1.1 for the definition.

- encoding_polynomial2: This 8 bit field specifies encoding polynomial 2. See clause A.3.1.1 for the definition.

- subblock_input_length: This 16 bit field specified the encoding input length for the subblock.

- subblock_output_length: This 16 bit field specified the encoding output length for the subblock. By definition, this value is larger or equal than the value of the subblock_input_length.

- interleaver_id: This 8 bit field identifies the interleaver defined in ICT (interleaver configuration table) to be used.

- uw_length: This is an 8 bit field specifying the UW length in bits. The loop which follows is aligned however on byte boundaries. This means that for example if the UW length is 14 bit, the loop over the UW bytes will consist of 2 Byte with the last 2 bit being merely stuffing bits.

- uw_bit: This is a 1 bit field specifying a bit of the UW. As the UW is not scrambled, a proper sequence should be selected in order to comply with ETSI requirements concerning off-axis EIRP. The bits are listed in transmission order (first bit listed = first bit send on air interface).

- stuffing_bit: Since the UW description, interleaver PI are byte aligned, stuffing bits are present until the next byte boundary. The stuffing bits may take any value and should be discarded by the terminal.

- normalisation_sequence_length: This 3 bit field specifies the number of normalization symbols.

- nbr_UW_sequences: This 8 bit field specifies the number of UW sequences to be inserted in the bursts.

- uw_sequence_start: This 15 bit field provides the position (expressed in number of bits) of the first bit of the UW sequence within the burst. A value of zero means the first bit of the burst.

- normalisation_sequence_flag: This 1 bit flag when set (value = 1) means that the UW sequence consists of a UW segment and a preceding normalization sequence. When not set (value = 0) this means there is no preceding normalization sequence and the UW sequence only consists of the corresponding UW segment.

- uw_segment_length: This 8 bit field specifies the number of bits in the UW segment.
- normalisation_sequence: This 8 bit field contains the entries of the normalization sequence lookup table. If the normalization_sequence_length in symbols multiplied by the log2(alphabet size M) is less than 8 bit the normalization sequence bits will start at the left most bit in the field and will be followed by a sufficient number of stuffing bits set equal to zero. The lookup table will always be signalled even in case no normalization is required.
- tx_block_size: The number of consecutive BTU's required for transmission of the physical layer block used by the specific tx type. For TDMA this indicates the size of the timeslot required for the burst, given in BTU's. For TDM this parameter is irrelevant and it is set to zero.
- tx_start_offset: For TDMA this is the nominal offset from the start of the timeslot in NCR ticks. (For TDM this parameter indicates the time in the future the transmission may start.

A.3 Return Link -Transmission Bursts

This clause defines the burst generation according to the generic CPM burst structure.

A.3.1 Generic Coding and Interleaving

A.3.1.1 The Generalized Sub-Block Polynomial Encoder

The Generalized Sub-block Polynomial Code (GSPC) is defined by FEC parameters signalled to the terminal in the BCT Table. Main principles of the coding scheme are defined below.

The rate *K*/*N* Generalized Sub-block Polynomial encoder segments the uncoded *K*-bit data sequence delivered by the CRC encoder,

$$X = \left[x_{K-1}, x_{K-2}, \cdots, x_0\right]$$

in N_b blocks sub-blocks with length k_1, k_2,…k_{Nb}.

$$\sum_{j=1}^{N_b} k_j = K$$

$$X^{(j)} = \left[x^{(j)}_{k_j-1}, x^{(j)}_{k_j-2}, \cdots, x^{(j)}_0\right]$$

$$x^{(j)}_m = x_{m'(m,j)}$$

$$m'(m,j) = m + \sum_{j'=j+1}^{Nb} k_j$$

$$\begin{cases} j < q_1 \Rightarrow k_j = k_{shortened} + 1 \\ j \geq q_1 \Rightarrow k_j = k_{shortened} \end{cases}$$

$$\begin{cases} j < q_1 \Rightarrow n_j = n_{shortened} + 1 \\ j \geq q_1 \Rightarrow n_j = n_{shortened} \end{cases}$$

Each sub-block is encoded using a binary systematic or non-systematic linear polynomial or convolutional code (signalled by the FEC parameters).

FEC variant 1

In case of FEC variant 1, the encoder appends a number of parity check bits equal to the degree of the encoding polynomial for the *j*th block. It also has the option to add an overall parity bit. Let us denote the output bits of the jth sub-block as:

$$Y^{(j)} = \left[y^{(j)}_{n_j-1}, y^{(j)}_{n_j-2}, \cdots, y^{(j)}_0\right] = \left[x^{(j)}_{k_j-1}, x^{(j)}_{k_j-2}, \cdots, x^{(j)}_0, p^{(j)}_{d_j-1}, \cdots, p^{(j)}_0\right]$$

where $n_j - k_j = d_j$

$$\sum_{j=1}^{N_b} n_j = N$$

The mathematical definition of the sub-block polynomial encoder is:

$$Y^{(j)}(D) = Y^{(j)}_{reg}(D) \cdot D^{e^{(j)}} + e^{(j)} \cdot Y^{(j)}_{reg}(1)$$

$$Y^{(j)}_{reg}(D) = X^{(j)}(D) \cdot D^{d_j} + P^{(j)}(D)$$

where the binary polynomials *X*, *Y*

$$X^{(j)}(D) = \sum_{t=0}^{k_j-1} x^{(j)}_t D^t$$

$$Y^{(j)}_{reg}(D) = \sum_{t=0}^{n_j-1-e^{(j)}} y^{(j)}_t D^t$$

$$Y^{(j)}(D) = \sum_{t=0}^{n_j-1} y^{(j)}_t D^t$$

$$P^{(j)}(D) = \sum_{t=0}^{d_j-1-e^{(j)}} p^{(j)}_t D^t$$

And where the generator polynomial is defined as:

$$G_S(D) = \sum_{t=0}^{d_j-e^{(j)}} g_t D^j$$

The 'parity' polynomial is obtained as the *remainder* of the division as defined in:

$$P^{(j)}(D) = X^{(j)}(D) \cdot D^{d_j - e^{(j)}} \bmod G_S(D)$$

Note that the code reduces to a regular polynomial code if the $e^{(j)}$ bit is set to zero. If the $e^{(j)}$ bit is set to one, an overall parity bit is added to the polynomial code.

FEC variant 2

FEC variant 2 is a non-systematic binary convolutional code. The encoder generates the following codeword:

$$Y^{(j)}(D) = G_{NS1}(D) \cdot X^{(j)}(D) + G_{NS2}(D^2) \cdot X^{(j)}(D^2) \cdot D$$

where the generator polynomials are given by the expression below for i=1,2:

$$G_{NSi}(D) = \sum_{t=0}^{d_j} g^{i}_{d_j - t} D^t$$

Note that d_j has the meaning of polynomial degree, here (and not the amount of parity bits).

Practical implementation

Note that both encoders (systematic and non-systematic) can be implemented using the same circuit. A possible implementation is shown below.

The canonical implementation of both schemes is a shift register with feedback (that can be enabled or disabled) starting in the all-0 state. In case of the systematic code, feedback is enabled and the feedback taps are equal to the generator polynomial of the polynomial code.

In case of a non-systematic code, feedback is disabled and the filter taps are equal to the generator polynomial of the convolutional code. Note that for the non-systematic encoder, this circuit should be run twice (once for each polynomial).

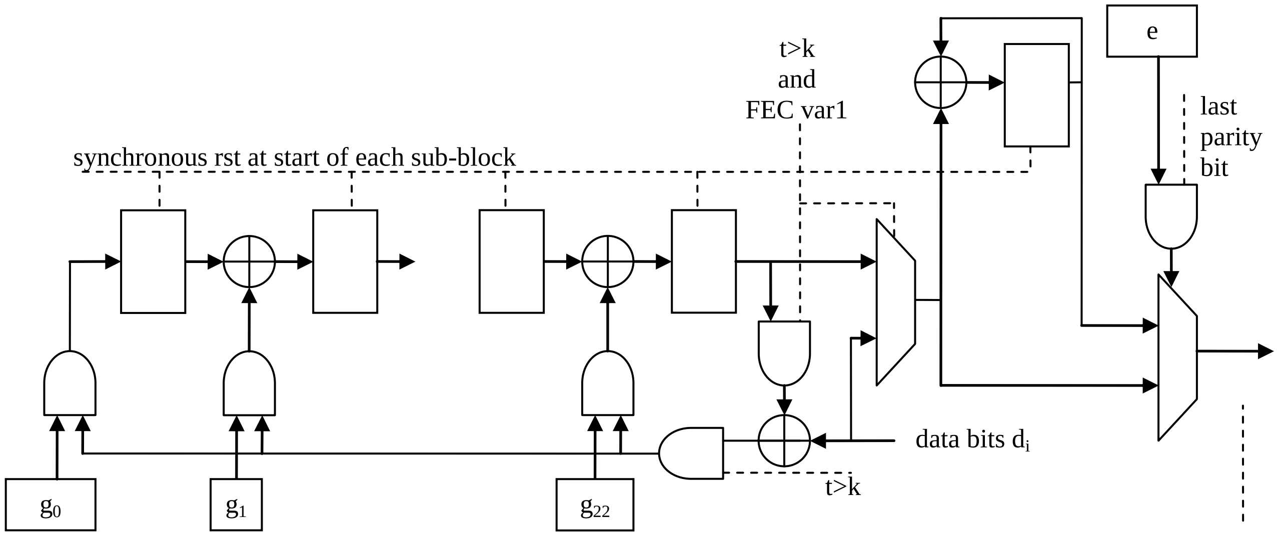


Figure A.3: Practical implementation

Signalling

The number of sub-blocks N_b, the generic channel interleaver (see below) and the code type (systematic / non-systematic) are signalled by Table BCT. In addition, for each sub-block and for each transmission mode, the encoding polynomials G and the parameters k_j, d_j and e_j are indicated by Table BCT.

The range of all these parameters and the mapping to the signalling table from section is defined in the signalling section.

Table A.11: Mapping parameters to Signal Name

Parameter	Signal name
FEC variant	encoder_type
N_b	number_of_subblocks
e_j	subblock_parity
G_S or G_{NS1}	encoding_polynomial 1
G_{NS2}	encoding_polynomial 2
$k_{shortened}$	subblock_input_length
$n_{shortened}$	subblock_output_length
q_1	one_bit_longer_subblocks

The two polynomials are defined by 23 and 8 bits respectively. The first bit g_0 is always 1. Mapping on the signalled bits is defined in the table below. The first bit in the signalled sequence is denoted by x_1.

Polynomial 1	$g_0=x_1,\ldots,g_{22}=x_{23}$
Polynomial 2	$g_0=x_1,\ldots,g_7=x_8$

In case of FEC variant 1, there is only one polynomial and the second polynomial definition is ignored.

A.3.1.2 Generic Interleaver and Puncturer

The interleaver should be totally programmable and is therefore implemented as lookup tables, which are signalled in the BCT Table. The look-up table of the interleaver defines the mapping between the position of a bit in the coded data block entering this interleaver and the position of this bit in the data block leaving the interleaver.

The look-up table defines the function n(m), with index m = 0,1 …, M-1 and values n = 0, 1, ... , N-1, whereas N is the size (number of bits) of the coded data block and M is the size of the coded data block after (optional) puncturing. Value n = 0 corresponds to the first bit entering the interleaver, index m = 0, corresponds to the first bit leaving the interleaver.

The look-up table will be signalled as M words of 12 bit, whereas each word represents a value n for the corresponding index m. The word for index m = 0 is sent first.

A.3.2 Generic Continuous Phase Modulation

The CPM modulation is defined through the following parameters (programmable and signalled in the Table BCT):

- alphabet size, *M:*
  - values *M*= 2, 4, 8
- modulation index, *h:*
  - the modulation index *h* is rational fraction *h* = *m*/*p* <=1 with *m* and *p* relative prime positive integers in the range [1,2,…,8].
- phase pulse samples, *q*(*t*):
  - generic with fixed duration L=2 symbols:
    - q(t)= $\begin{cases} 0, t < 0 \\ \int_0^t g(\tau)d\tau, 0 \le t \le LT_s \\ 0.5, t > LT_s \end{cases}$, where L is the memory length of the CPM modulation(L=2)
    - $g(t)$ is the frequency response, $g_{RC}(t)$ and $g_{REC}(t)$ represent the raised-cosine(RC) and rectangular(REC) pulse.
    - $g_{RC}(t) = \begin{cases} \frac{1}{4T_s}(1-\cos\frac{\pi t}{T_s}), 0 \le t \le LT_s \\ 0, otherwise \end{cases}$ $\mathrm{g_{RC}(t)} = \begin{cases} \frac{1}{4T_s}\left(1-\cos\frac{\pi t}{T_s}\right), 0 \le t \le 2T_s \\ 0, \mathrm{otherwise} \end{cases}$ and

$$g_{RC}(t) = \begin{cases} \frac{1}{4T_s}, 0 \le t \le LT_s \\ 0, otherwise \end{cases} \quad \mathrm{g_{REC}(t)} = \begin{cases} \frac{1}{4T_s}, 0 \le t \le 2T_s \\ 0, \mathrm{otherwise} \end{cases}$$

  - generic with fixed duration L=3 symbols:
    - Note this includes effective duration L_{eff}<=3 symbols
  - symmetrical representation $q(t)=1/2 - q(L-t)$
- symbol mapping method

The CPM waveform is expressed by the following formula:

$$\tilde{s}(t,\alpha) = \exp\left[j \cdot 2\pi h \sum_{n=0}^{+\infty} \alpha_n q(t - nT) \right]$$

The real transmitted signal (at carrier frequency f_0) is:

$$s(t,\alpha) = \mathrm{Re}\left\{ \tilde{s}(t,\alpha) e^{j2\pi f_0 t} \right\}$$

where

- T is the symbol period.
- α_n is the sequence of uncorrelated data information symbols to be transmitted and each taking one of the values {±1, ±3, ±(M-1)}.
- $q(t)$ is the phase pulse.

A.4 Reference waveforms for Generic CPM (eBCH-CPM scheme)

Table A.12 lists the reference waveforms for generic continuous phase modulation format class bursts considering eBCH-CPM scheme. The parameters follow the syntax specified in clause A.2.2.1.1.

Table A.12: Reference Waveforms for Generic Continuous Phase Modulation bursts (eBCH-CPM scheme)

Waveform id	Content Type	FEC input bit length (K)	FEC output bit length (N)	Preamble (uw) length	Data #1 bit length	Normalization _seq. #1	Midamble (uw) length	Data #2 length	Normalization _seq. #2	Burst symbol length	Alphabet size (M)	Modulation index (h)	Code rate	eBCH (subblock_output_length , subblock_input_length)	Number of subblock (number_of_subblocks)	Carrier Spacing	Spectral Efficiency	Memory length (L)	UW bits (uw_bits) (Preamble+Midamble)	Phase Response
1	Logon	456	768	64	64	6	64	704	6	454	4	4/7	0,591	(64,51)	24	2,3259	0,5	2	7CD593ADF7818AC8	Raised Cosine
2	Control	168	272	64	64	6	64	208	6	206	4	4/7	0,591	(64,51)	8	2,3259	0,5	2	7CD593ADF7818AC8	Raised Cosine
3	Traffic /Control	400	673	64	64	6	64	609	6	407	4	4/7	0,591	(64,51)	21	2,3259	0,5	2	7CD593ADF7818AC8	Raised Cosine
4	Traffic /Control	400	556	64	64	6	64	492	6	348	4	3/7	0,7053	(64,51)	12	1,8773	0,75	2	7CD593ADF7818AC8	Raised Cosine
5	Traffic /Control	400	556	64	64	6	64	492	6	348	4	2/7	0,7162	(64,51)	12	1,3766	1	2	7CD593ADF7818AC8	Raised Cosine
6	Traffic /Control	400	475	64	64	4	64	411	4	306	4	1/4	0,847	(128,113)	5	1,15	1,5	2	7CD593ADF7818AC8	Raised Cosine
7	Traffic /Control	400	460	64	64	6	64	396	6	298	4	1/5	0,871	(128,113)	4	0,99	1,8	2	7CD593ADF7818AC8	Raised Cosine
8	Traffic /Control	1 024	1 726	64	64	6	64	1 662	6	933	4	4/7	0,591	(64,51)	54	2,3259	0,5	2	7CD593ADF7818AC8	Raised Cosine
9	Traffic /Control	1 024	1 440	64	64	6	64	1 376	6	790	4	3/7	0,7053	(64,51)	32	1,8773	0,75	2	7CD593ADF7818AC8	Raised Cosine
10	Traffic /Control	1 024	1 427	64	64	6	64	1 363	6	784	4	2/7	0,7162	(64,51)	31	1,3766	1	2	7CD593ADF7818AC8	Raised Cosine
11	Traffic /Control	1 024	1 204	64	64	4	64	1 140	4	670	4	1/4	0,847	(128,113)	12	1,15	1,5	2	7CD593ADF7818AC8	Raised Cosine

Waveform id	Content Type	FEC input bit length (K)	FEC output bit length (N)	Preamble (uw) length	Data #1 bit length	Normalization _seq. #1	Midamble (uw) length	Data #2 length	Normalization _seq. #2	Burst symbol length	Alphabet size (M)	Modulation index (h)	Code rate	eBCH (subblock_output_length , subblock_input_length)	Number of subblock (number_of_subblocks)	Carrier Spacing	Spectral Efficiency	Memory length (L)	UW bits (uw_bits) (Preamble+Midamble)	Phase Response
12	Traffic /Control	1 024	1 174	64	64	6	64	1 110	6	657	4	1/5	0,871	(128,113)	10	0,99	1,8	2	7CD593ADF7818AC8	Raised Cosine
13	Traffic /Control	1 504	2 544	64	64	6	64	2 480	6	1 342	4	4/7	0,591	(64,51)	80	2,3259	0,5	2	7CD593ADF7818AC8	Raised Cosine
14	Traffic /Control	1 504	2 128	64	64	6	64	2 064	6	1 134	4	3/7	0,7053	(64,51)	48	1,8773	0,75	2	7CD593ADF7818AC8	Raised Cosine
15	Traffic /Control	1 504	2 089	64	64	6	64	2 025	6	1 115	4	2/7	0,7162	(64,51)	45	1,3766	1	2	7CD593ADF7818AC8	Raised Cosine
16	Traffic /Control	1 504	1 774	64	64	4	64	1 710	4	955	4	1/4	0,847	(128,113)	18	1,15	1,5	2	7CD593ADF7818AC8	Raised Cosine
17	Traffic /Control	1 504	1 714	64	64	6	64	1 650	6	927	4	1/5	0,871	(128,113)	14	0,99	1,8	2	7CD593ADF7818AC8	Raised Cosine

A.5 CPM Complexity

This clause provides an overview of algorithmic complexity of implementing CPM scheme at the transmitter and the receiver.

A.5.1 eBCH-CPM Modulator

A.5.1.1 eBCH Encoder

Figures A.4 and A.5 depict the functional operation flow in the process of eBCH encoding. First, the 'Message' data is fed into eBCH Encoder controller block. The BCH parity can be generated by the waveform ID about K_short data input. The tail bit generator makes one tail bit and eBCH output controller constructs the entire eBCH_output data. The specific operation procedure is as follows:

1) According to the waveform ID, 'BCH Encoder Controller' selects BCH 'Generator polynomial Coefficients'.

2) According to the waveform ID, 'BCH Encoder Controller' sends to 'Parity_calulation' block by appending 'zero' bits, the number of N_zero to K_short message information.

3) While 'Message' data including the number of K(N_zero+K_short) is fed, it is open in gate1 and off in gate2 in 'Parity calculation block'.

4) The gate1 is off when the number of K, all 'Message' data is fed. As soon as gate2 becomes "on" mode, the parity bits outputs through shift register operation.

5) While calculating the parity in 'BCH Encoder Controller', 'Message' data outputs and 'Party bits' outputs, subsequently.

6) The tail bit should be generated through 'N_short-1' in eBCH output controller.

7) Finally,'eBCH_Out' (N_short) data bits are the outputs of the encoder.

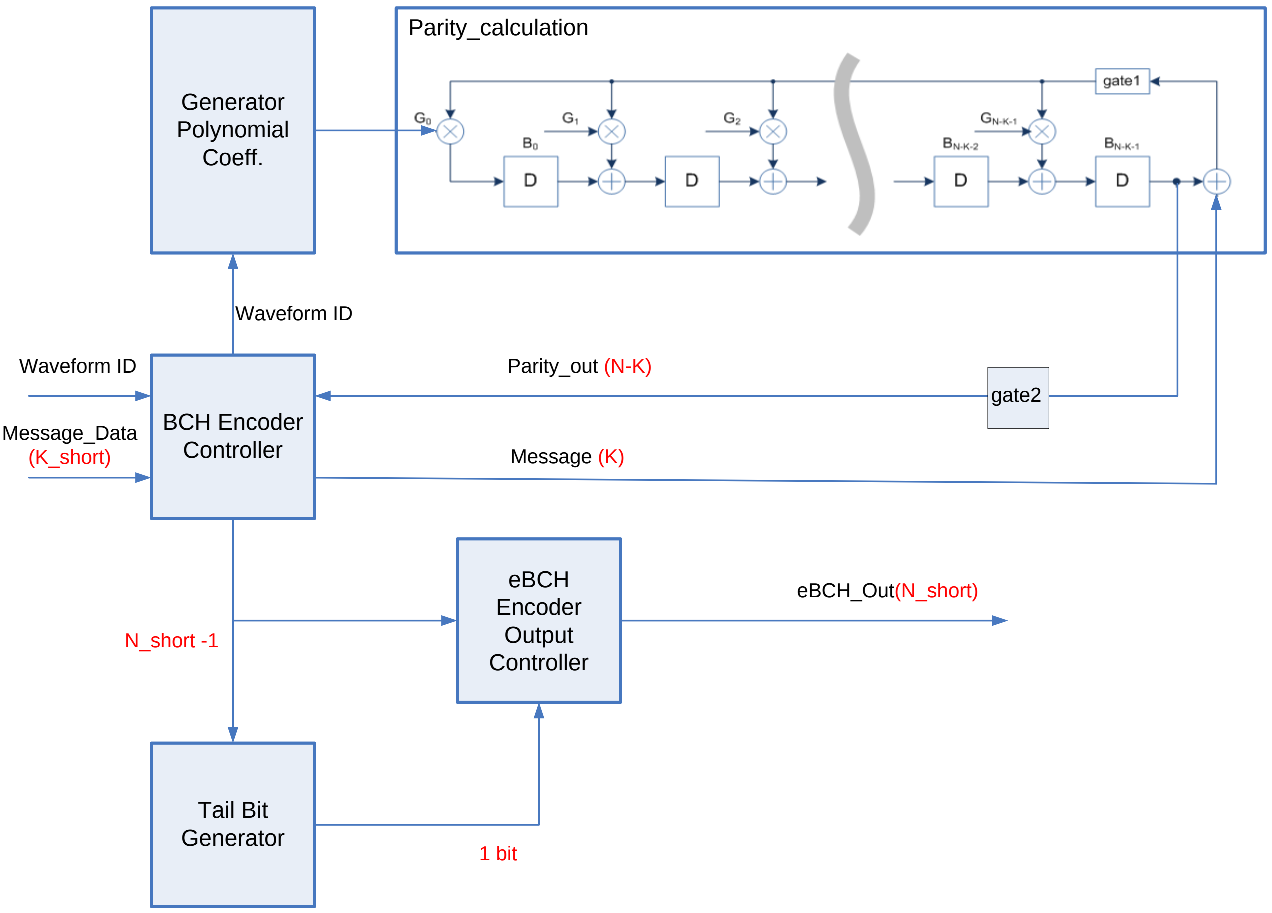


Figure A.4: eBCH encoder functional architecture

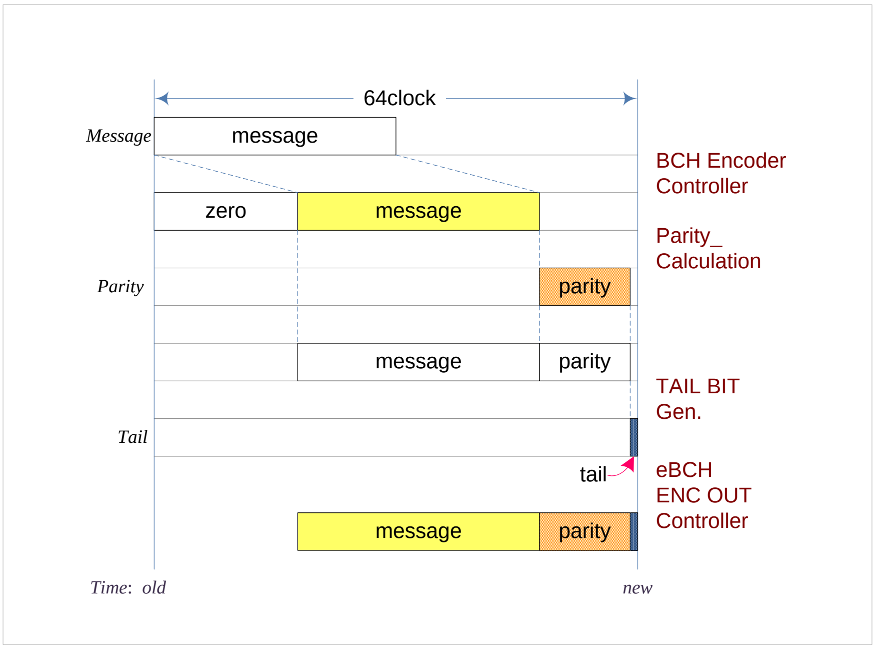


Figure A.5: Timing diagram of eBCH Encoder process

Table A.13 represents the generator polynomial coefficient according to the different waveform ID.

Table A.13: BCH Code Generator Polynomial

	G_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}	G_{13}	G_{14}
eBCH (51, 64)	1	0	0	1	1	1	0	0	1	0	1	0	1	-	-
eBCH (113, 128)	1	1	1	0	1	1	1	0	1	1	0	0	0	0	1

A.5.1.2 eBCH+CPM Modulator

Based on the proposed eBCH+CPM design, it has been implemented using an FPGA Xilinx XC5VLX110 model chip. The only core module such as eBCH encoder, S random interleaver, CPM modulator and digital up-convertor have been considered in the target FPGA. The other interface blocks such as PCI communication, CPU module, RLE, DAC(Digital Analogue Convertor) and forward link signalling parsing have not been considered to compute complexity. In the modulator aspects, LUT amount used for S random interleaver design has been occupied when it has been implemented as tabularized approach.

Table A.14: FPGA utilization

Slice logic Utilization	Utilization (Used/Total)
Number of Slice Register	2 % (1420/69120)
Number of Slice LUTs	1 % (506/69120)
Number of occupied Slices	2 % (504/69120)
Number of fully used LUT-FF pairs	16 % (279/1647)

A.6 CPM mobile performance

This clause illustrates the applicability of generic CPM modulation to mobile environment. In particular a channel model based on Rice-distributed fading is assumed.

Software simulations have been carried out under the assumption of highly correlated Doppler spectrum by considering directive antenna deployment. Two cases are envisaged, depending on the directivity of the antenna, which can be linked to both Rice factor and normalized Doppler frequency.

Specifically, higher antenna directivity values correspond to higher Rice factors and to lower normalized Doppler frequencies. To investigate the effect of mobile channel on the CPM waveform, an ideal channel estimation at the receiver. Table A.16 presents the simulation results, comparing AWGN and Mobile channel. As shown, for mobile LOS (K=17 dB, Rice Factor), more than 2 dB loss should be considered compared the results obtained with the ideal AWGN channel.

Table A.15: Simulation Parameters

Parameter	Value(s)		
Spectral Efficiency [bit/s/Hz]	0,5	0,75	1
Modulation Index	4/7	3/7	2/7
Modulation Order ($\log_2 M$)	2		
Target Code Rate	0,591	0,7053	0,7162
Code Block size [bit]	1 024		
Baud Rate [kbaud]	512		
Rice Factor K [dB]	17		
Speed [km/h]	150		
Normalized Doppler Frequency	8×10^{-4} (Ku:14GHz, Ka:30GHz)		

Table A.16: eBCH+CPM Thresholds (PER = 10^{-3})

Spectral efficiency	AWGN	Mobile(K=17)	Gap(degradation)
0,5	1,8 dB	4,1 dB	2,3 dB
0,75	2,1 dB	4,5 dB	2,4 dB
1	3,15 dB	5,4 dB	2,25 dB

Annex B: Examples of CRDSA Implementation and Performance

This annex provides some examples of performance and implementation results for the CRDSA technique [i.57], [i.58] applied to random access return channels based on software simulations and a hardware implementation which has served as a testbed to validate this technology [i.60].

B.1 Introduction

Slotted Aloha (SA) protocol [i.55] or enhanced version of the scheme, such as Diversity Slotted Aloha (DSA) [i.56], are used in TDMA systems with low efficiency and reliability typically for logging into the network. Recently an enhanced version of DSA dubbed Contention Resolution Diversity Slotted Aloha (CRDSA) has been introduced [i.57]. The CRDSA key idea is to transmit two or more (total) replicas of the same packet at random locations within the same frame as in DSA but with a slightly higher signalling to point to the location of other packet replicas. In case of successful packet reception at the gateway this extra signalling information allows to locate the twin packet within the frame and to accurately cancel it in addition to the one successfully decoded. To be remarked that using a powerful Forward Error Correcting (FEC) code and an adequate signal to noise ratio, there is probable to correctly detect the packet even in the presence of other colliding bursts. By iterating the process a number of times some of the initially lost packets can be recovered, improving the DSA performance. Results reported in [i.58] and [i.59] indicate that the CRDSA throughput is significantly higher than that of conventional Slotted Aloha and Diversity Slotted Aloha. The throughput gain of the CRDSA compared to conventional random access scheme is mainly due to successive interference cancellation. The CRDSA performance improves noticeably with burst power unbalance (received from different user terminals within the same time slot). Such power unbalance facilitates the successive cancellations performed by the burst demodulator.

The high throughput/high reliability CRDSA technique in a random access scenario makes this scheme suitable to support a low-duty cycle bursty traffic such as VSAT and broadband access return link signalling packets as well as SCADA applications.

B.2 CRDSA Parameter Settings Example

In this clause some examples on how to get the best performance of the CRDSA TDMA random access scheme are provided. Examples are based on relevant literature or unpublished results from which some key results have been extracted. Simulation results reported in this sub-section are based on the 3GPP FEC coding scheme [i.64] and using a burst containing100 information bits. Each frame is composed of 100 time slots and the traffic is assumed to have a Poisson distribution. The number of slots per frame should be kept possibly above 60 since lowering the number of slots per frame may degrade the CRDSA performance. In an MF-TDMA system, CRDSA time slots can be distributed in a two-dimensional time/frequency plane, for example using 10 time slots combined with 6 frequency bins (a total of 60 time slots). It should be noted that a different selection of the coding scheme may impact the performance results, however the same trend as a function of other system parameters are expected.

In the following simulated scenarios, the performance is measured in terms of the maximum normalized throughput and the packet Loss Ratio (PLR) as a function of the system traffic loading. The performance results are obtained in the presence of AWGN with a noise density corresponding to a link quality of $E_s/N_0 = 10$ dB (excluding the signal power fluctuation). In addition, signal power fluctuations (among the bursts received from different RCSTs) are lognormally distributed with zero mean and parameterized standard deviation σ (measured in dB).

B.2.1 Performance dependence on the number of replicas

It is important to assess the impact of the number of burst replicas on the CRDSA performance. Results shown in Figure B.1, indicate that at a target PLR of 10^{-3}, the best performance is obtained with four replicas. The steepness of the PLR characteristic versus the MAC load is increasing with the number of replicas but using more than 4 replicas reduces the throughput (measured at the target PLR = 10^{-3}).

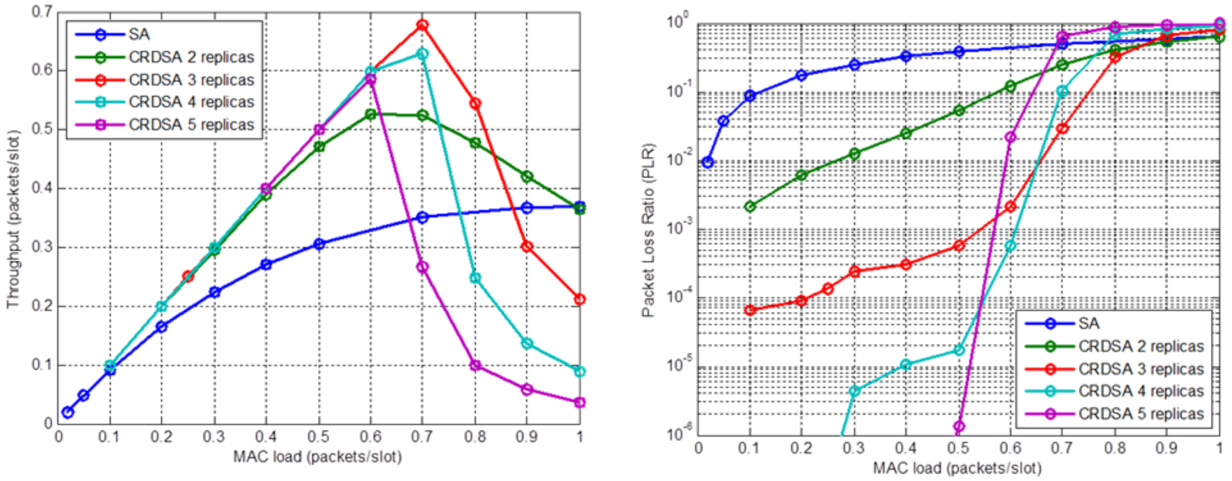


Figure B.1: CRDSA performance versus the number of replicas, 3GPP FEC coding rate ½, E_s/N_0 = 10 dB, equipowered burst

B.2.2 Performance dependence on the burst power unbalance

Figure B.2 shows the CRDSA performance assuming that bursts' power is randomly distributed according to a lognormal distribution with zero mean and parameterized standard deviation σ. It can be observed that the MAC throughput improves reaching a value above 1 for σ = 3 dB (Throughput values above 1 is possible thanks to the interference cancellation). At the same time a PLR floor appears for σ = 3 dB due to the non-negligible probability that the received packet power is too low to avoid packet errors in the presence of AWGN. This effect can be mitigated by increasing the operating E_s/N_0, reducing the FEC code rate (see clause B.2.3) or limiting the value of σ. It should be noted that the assumption of lognormal burst power distribution may be pessimistic as in practice system power fluctuations can be better represented by a truncated lognormal distribution that mitigates the PLR floor effects.

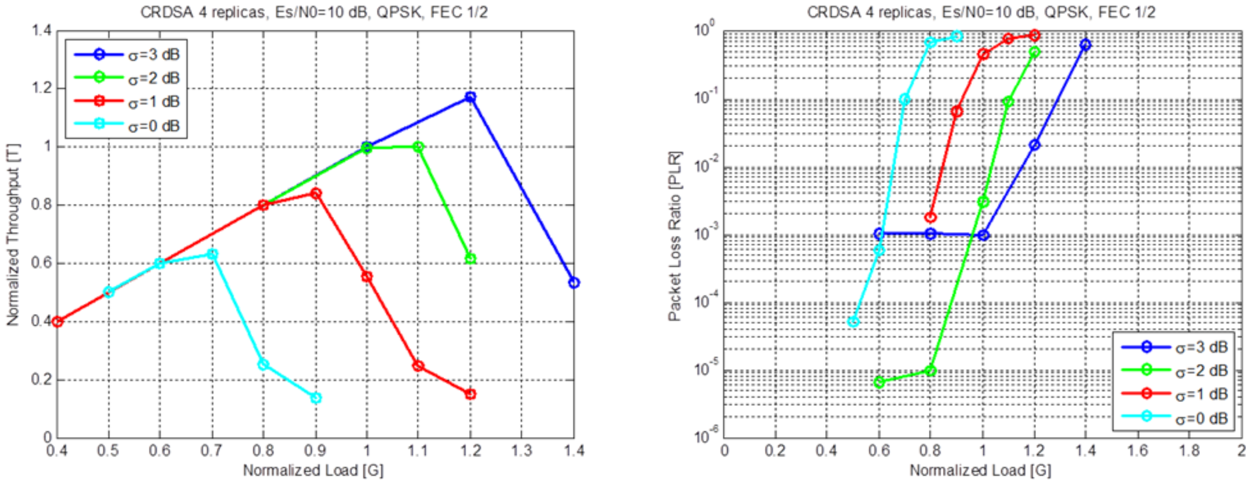


Figure B.2: CRDSA performance versus the burst power unbalance, 3GPP FEC coding rate ½, E_s/N_0 = 10 dB, lognormal burst power distribution

B.2.3 FEC Coding rate impact

A final aspect to be assessed is the impact of the FEC coding rate on the CRDSA performance. Also in this case the results are counterintuitive as being TDMA an orthogonal access scheme one can expect that the highest throughput is obtained with the highest FEC coding rate. Instead being interested in the aggregate MAC random access throughput the results reported in Figure B.3 indicate that the best performance is obtained for r = 1/3. The throughput is similar to that of r = 1/2 but the floor for σ = 3 dB is now reduced because of the better FEC performance. Note that normalized load refer to the information bits and not to the coded symbols to allow making a fair comparison.

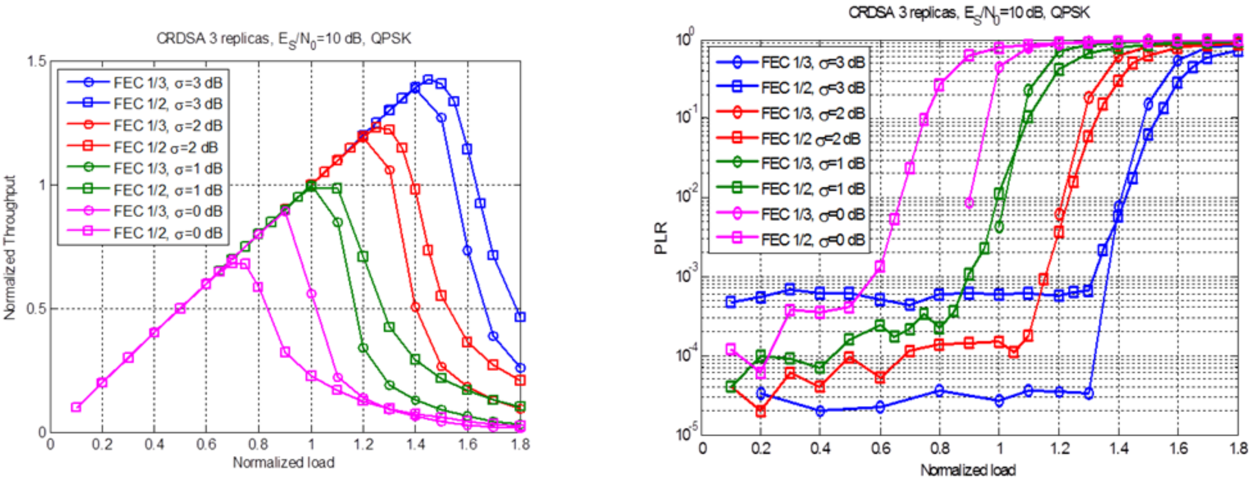


Figure B.3: CRDSA performance versus the burst power unbalance, 3GPP FEC coding rates ½ and 1/3, E_s/N_0 = 10 dB, lognormal burst power distribution

B.3 Reference CRDSA Implementation

This clause describes the hardware implementation, configuration and the lab performance results of a reference design of CRDSA scheme [i.60].

B.3.1 CRDSA Frame Structure

The main configuration parameters used to validate the CRDSA access scheme in [i.60] are reported in Table B.1.

Table B.1: Hardware Configuration Parameters for CRDSA Implementation

Parameter	Value
Modulation ♦ Coding Rate	QPSK ♦ 1/2
Payload (bits)	488
Pre-amble ♦ Post-amble ♦ Pilots ♦ Guard (symbols)	40 ♦ 12 ♦ 54 ♦ 6
Symbol Rate (k symbols/sec)	128
Burst Length (symbols) ♦ Burst Duration (msec)	600 ♦ 4.69
Bit Rate (kbits/sec)	104.11
Bursts per Frame ♦ Frame Duration (msec)	66 ♦ 309.38
Number of Replicas (including original) per frame	4
Max Number of overlapping Bursts	8

Each CRDSA frame consists of 66 time-slots located on a single carrier where the original bursts and their replicas are allocated. The number of replicas (including the original burst) is four in each CRDSA frame. Each burst is 600 symbols long, including preamble and pilot symbols and the adopted modulation and coding rate are respectively QPSK and 1/2. The MODEM works with a baud rate of 128 ksymbols/sec and the demodulator is able to perform up to seven iterations to erase the replicas within a given time-slot.

The maximum number of overlapping bursts within a given time-slot is limited to eight bursts. The underlying coding used in the testbed is according to the DVB-RCS2 specifications (16-states duo-binary Turbo-ϕ Coding) [i.1].

The position of the replicas inside a given frame is obtained through running a Pseudo-Random Generator (PRG) for a sufficient number of times. The 8-bits seed of this PRG, sent to the receiver together to the 16-bits Terminal ID, is exploited by the receiver to re-generate the position of the replicas. Once a burst has been correctly demodulated (i.e. the CRC is found to be correct) it can be regenerated and subtracted from the time-slots where its replica(s) are located (see clause B.3.2 for more details).

Successive interference cancellation iterations over the whole frame are necessary to eliminate most of the burst replicas present. In a simplified scenario used for better illustrating the iterative cancellation approach the process starts with the detection of a "clean" burst (no overlapping bursts such as B1 as shown in Figure B.4), continue cancelling the demodulated bursts up to it is possible, end when no more cancellation can be done (e.g. because a burst collision, as in the case of bursts B7 and B8 in Figure B.4). In reality, because of the powerful FEC adopted or possible received bursts power unbalance, packets can also be detected in the presence of one or even more colliding bursts. So in general any detected burst (successful CRC check) is then removed together with its replica(s) from the frame being processed.

At the receiver, the replica to be subtracted is re-generated by the demodulator using the whole burst decoded and re-encoded data to perform maximum likelihood channel parameter estimation. In particular, since the replicas of a given burst are transmitted by the same modulator and being the frame duration limited in time, it can be reasonably assumed that they have the same amplitude, timing and frequency inside a frame. Therefore the re-generation can be done through using the timing and the frequency parameters estimated on the slot where the first clean burst of a given set of replicas was detected. This is not true for the replica's carrier phase which need to be re-estimated being rapidly time variant because of carrier phase noise or Doppler effects. This operation is carried out on the whole burst belonging to the slot where the replica should be subtracted through cross-correlating it with its already demodulated replica burst. Exceptionally this process may be required also for the burst amplitude variation if it is rapidly fluctuating due to fading effects. More details on channel estimation for CRDSA can be found in [i.61] and [i.62].

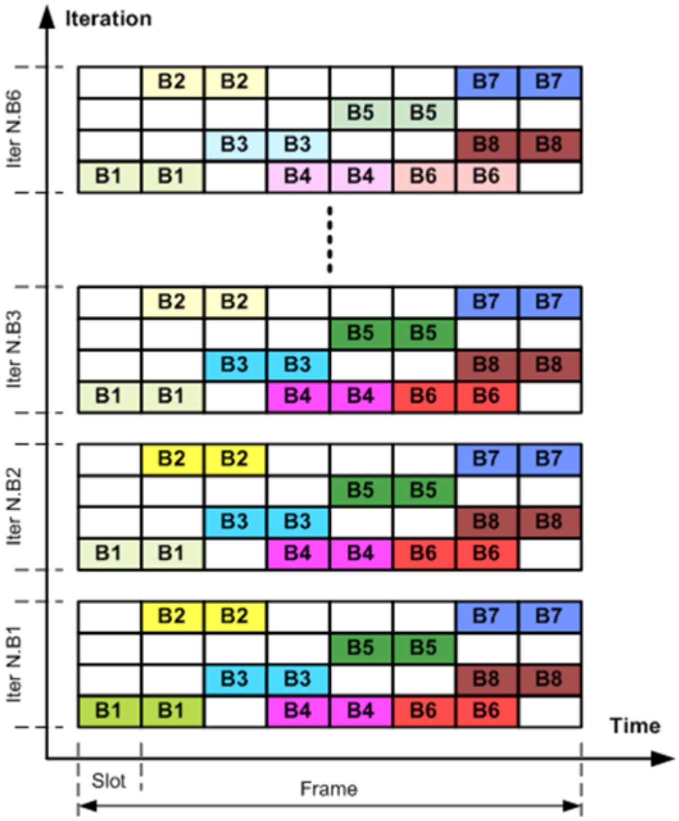


Figure B.4: Successive Interference Cancellation Process

B.3.2 MODEM Architecture

This clause describes the implementation of the modulator and demodulator in the reference testbed supporting CRDSA capability [i.60]. The transmitter baseband physical layer functionality is shown in Figure B.5. Compared to a conventional modulator, the CRDSA modulator uses frame memory to allow bursts' replication before transmitting. It can be tailored according to the maximum number of slots per frame (S < 66). The factor S determines the memory overhead required for supporting the CRDSA mechanism. As shown in Table B.1, S=66 is assumed.

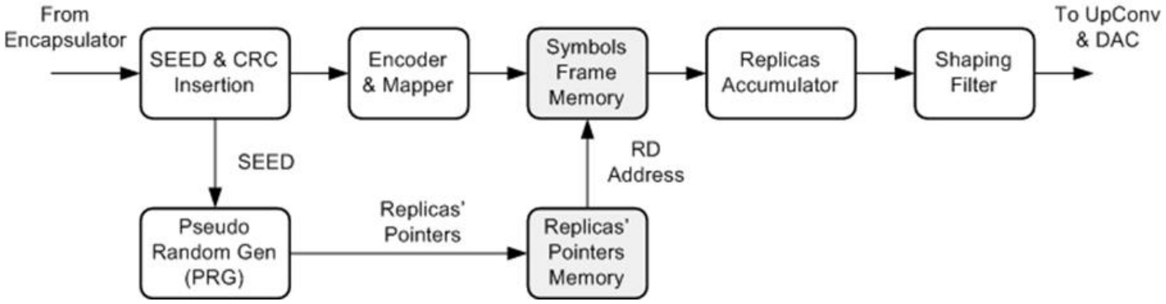


Figure B.5: CRDSA Modulator

Since the frame memory is typically based on a ping-pong (PP) implementation, exploiting 1 bit/symbol representation, this overhead can be estimated as:

$$H_{CRDSA-MOD} = S \cdot 2_{IQ} \cdot 2_{PP} \cdot 600_{SYMBS} = 158\,400\, bits$$

Where 600 symbols are considered in each transmitted burst as per Table B.1.

Each burst is labelled with the SEED used to initialize the PRG which provides the addresses of the bursts' replicas. When the frame memory has been filled with the original bursts, it is read back accumulating those allocated to the same slot, up to a maximum of eight bursts. It should be noted such accumulation step is not necessary in a realistic RCST and is merely required as part of the testbed functionality (single transmitter device).

The typical resources available in hardware (such as FPGA) are memories (usually organized in blocks of RAM), flip-flops, embedded multipliers and combinatorial logic implemented through look-up tables. The hardware implementation of the transmitter in the testbed indicates a maximum of 100 % increase in the memory usage in RCST, considering the frame size of 66 time slots.

Figure B.3 illustrates a functional diagram of the CRDSA receiver (Physical Layer base-band elements). The demodulating process is arranged into seven, double step iterations, correlating and cancelling interference within each frame.

The following proceeding steps are carried out in the demodulator:

- During the first step of the first iteration all the bursts of a frame are processes and those successfully demodulated are stored into the "Symbol Frame B Memory" (as shown in Figure B.6). At the same time a complete image of the frame is stored into the "Samples Frame Memory" and into the "Symbol Frame A Memory".
- In the second step of the first iteration the demodulator carries out the cross correlation between the A and B Symbol Frame Memories filling the "Synch Data Frame Memory" which is used to update the "Samples Frame Memory" cancelling from it the bursts demodulated during the first step.
- In the second iteration this double step procedure starts again and the cleaned content of the "Samples Frame Memory" is demodulated as far as it is possible storing the results in the A and B Symbol Frame Memories and repeating the same process carried out during the previous iteration.
- This procedure is repeated up to a pre-defined maximum number of iterations (set to 7 in the testbed configuration).

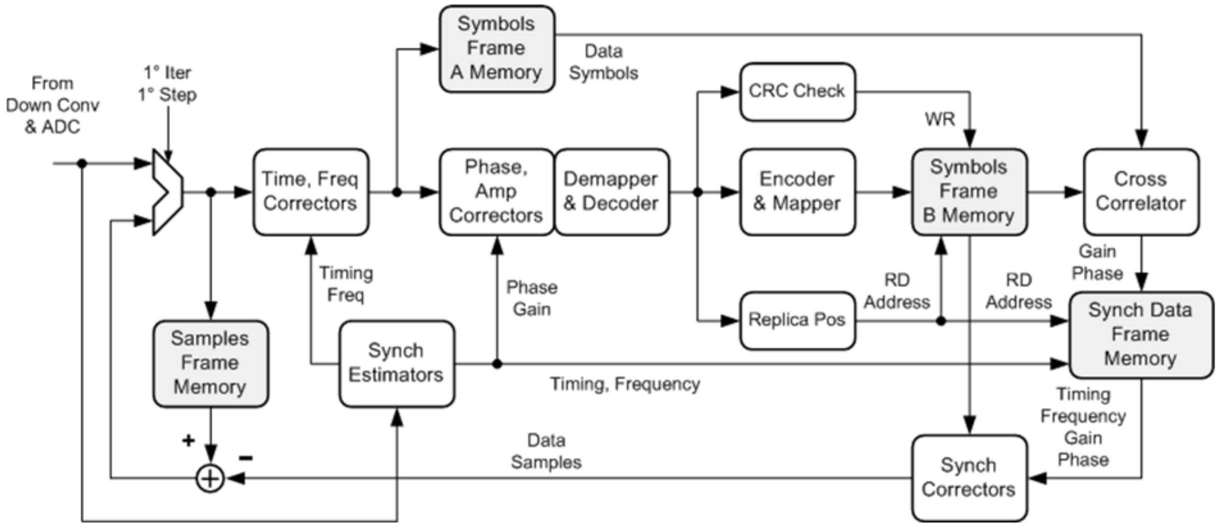


Figure B.6: CRDSA Demodulator

The complexity overhead of a CRDSA capable baseband demodulator, with respect to a standard DVB-RCS demodulator, can be investigated considering the two main architecture extensions required to support the cancellation process:

- the insertion of frame memory buffers,
- the execution of a given number of iterations.

The first item increases the quantity of memory (i.e. the employed silicon area), the latter increases the processing speed requirement (in order to achieve a given throughput). A higher processing speed may require parallelisation of running processes in order to cope with technological limitation of hardware.

The implemented operating mode (see Table B.1) allows improving the speed of the structure to support up to four times the real-time iterations, hence D=2 demodulators are necessary to support seven iterations. Actually the D demodulators are employed in frame division fashion: each one executes seven iterations working on one frame each. In order to assess the complexity, we consider bursts of 600 symbols and frames of S=66 slots and taking into account the memories shown in Figure B.3 together with the following assumptions:

i) Samples Frame Memory, complex 12 bits wide 4 samples per symbol;

ii) Symbols Frame A Memory, complex 12 bits wide;

iii) Symbol Frame B Memory, complex 1 bit wide; and

iv) Synch Data Frame Memory having a negligible complexity.

Overall complexity increase of the CRDSA demodulator (in terms of memory usage), compared to a conventional demodulator is around 4 times in this specific implementation.

This estimate does not include the parts that are in common with a normal MF-TDMA burst demodulator like the digital front-end and the frequency demultiplexer.

B.3.3 Performance Results

The behaviour of the implemented CRDSA demonstrator has been assessed based on the configuration reported in Table B.1 and under different assumptions concerning the users' power distribution. The traffic is assumed to be Poisson distributed but the maximum number of burst simultaneously transmitted is limited to 8 because of the modulator implemented limitations.

Figures B.7 to B.12 show the throughput achieved, in terms of correctly demodulated bursts per frame and the associated burst packet error rate (PER). Performance results obtained from the prototype hardware closely follow the software simulation results, as illustrated in each figure.

As it is known from literature results [i.59] and the experimental results shown in Figures B.7, B.9 and B.11, the power unbalance between the transmitted burst replicas can considerably improve the system throughput.

It is worth noting that overall performance of the CRDSA in terms of the achievable throughput, among other factors, depends on the error correction capability of the underlying FEC scheme, particularly in the presence of interference. This can be observed by comparing the performance results shown in figures below for Turbo-phi code compared to those reported in Figure B.3 for 3GPP code.

Overall, it is expected that an optimization of the FEC code may further improve the CRDSA throughput which even with the existing FEC designs already offers a significant improvement compared to conventional random access scheme such as SA or DSA.

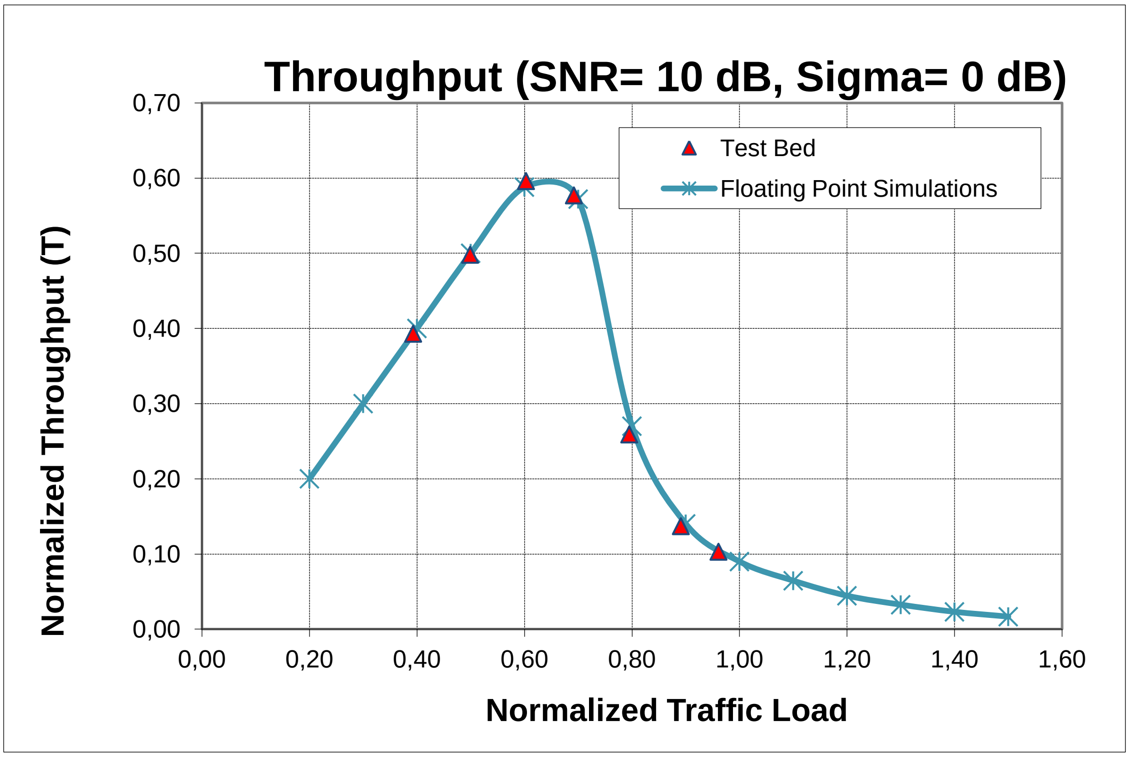


Figure B.7: Physical Layer Throughput (Bursts with Power Imbalance among the received bursts)

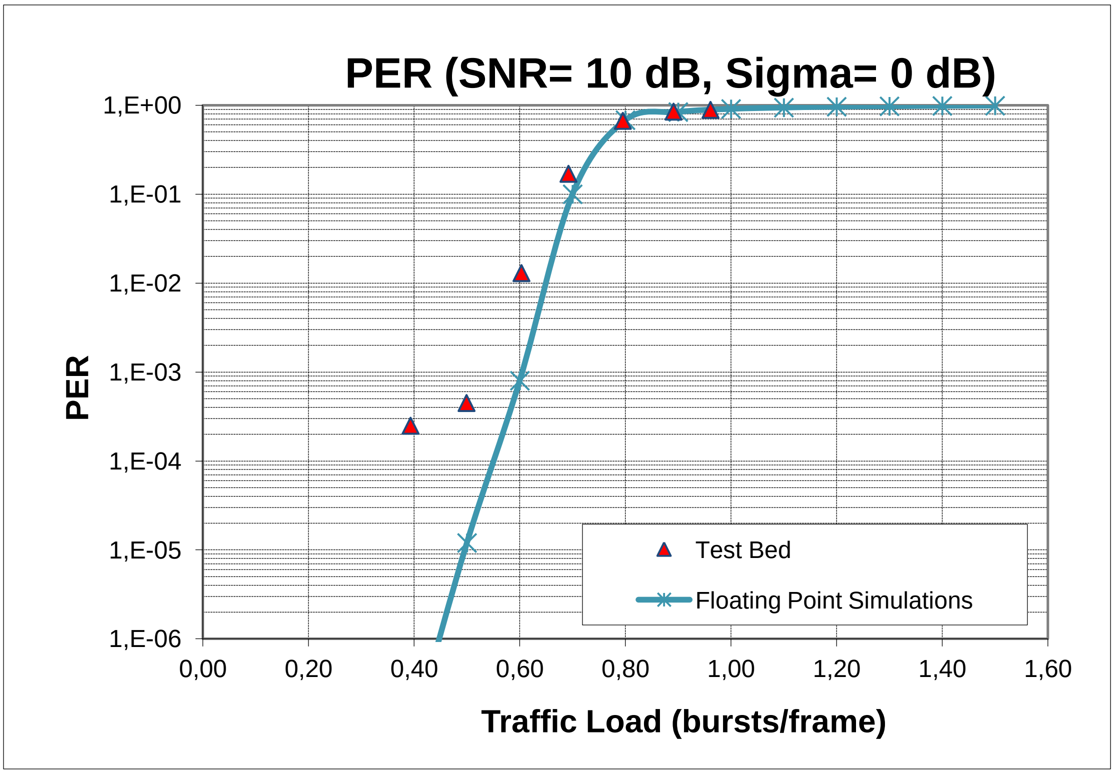


Figure B.8: Physical Layer Packet Error Ratio

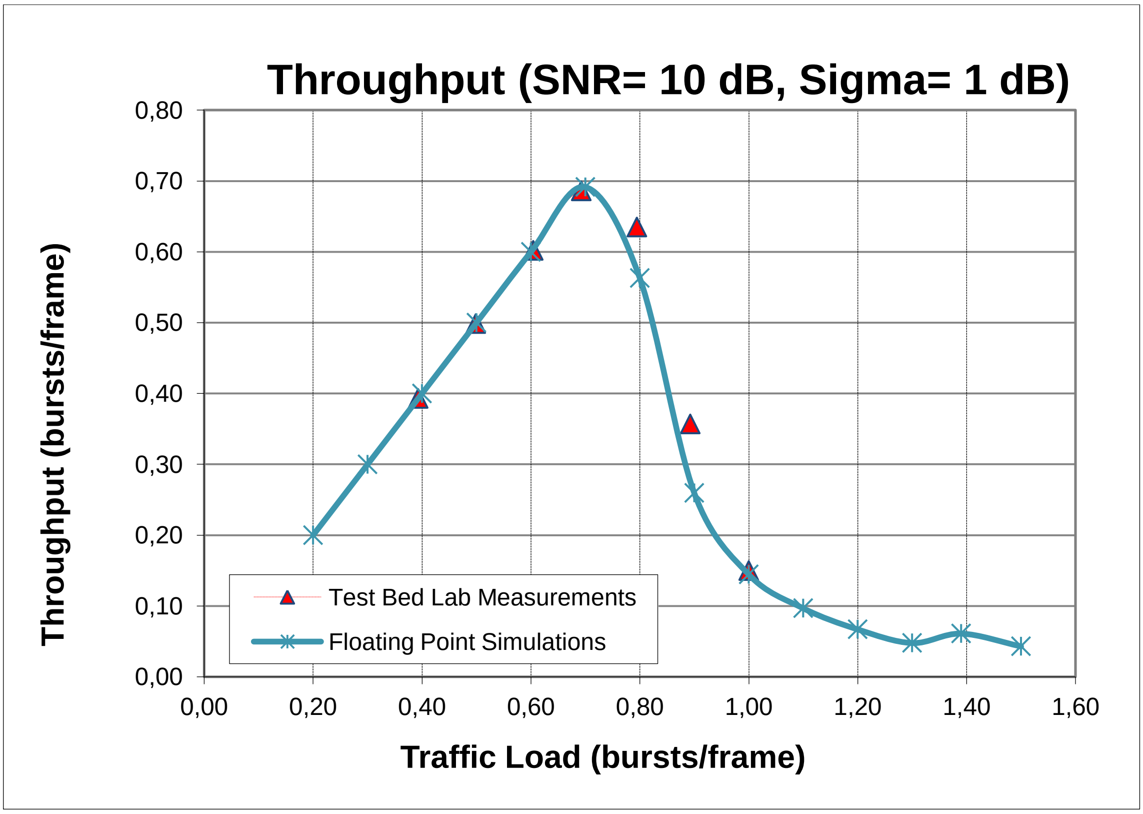


Figure B.9: Physical Layer Throughput

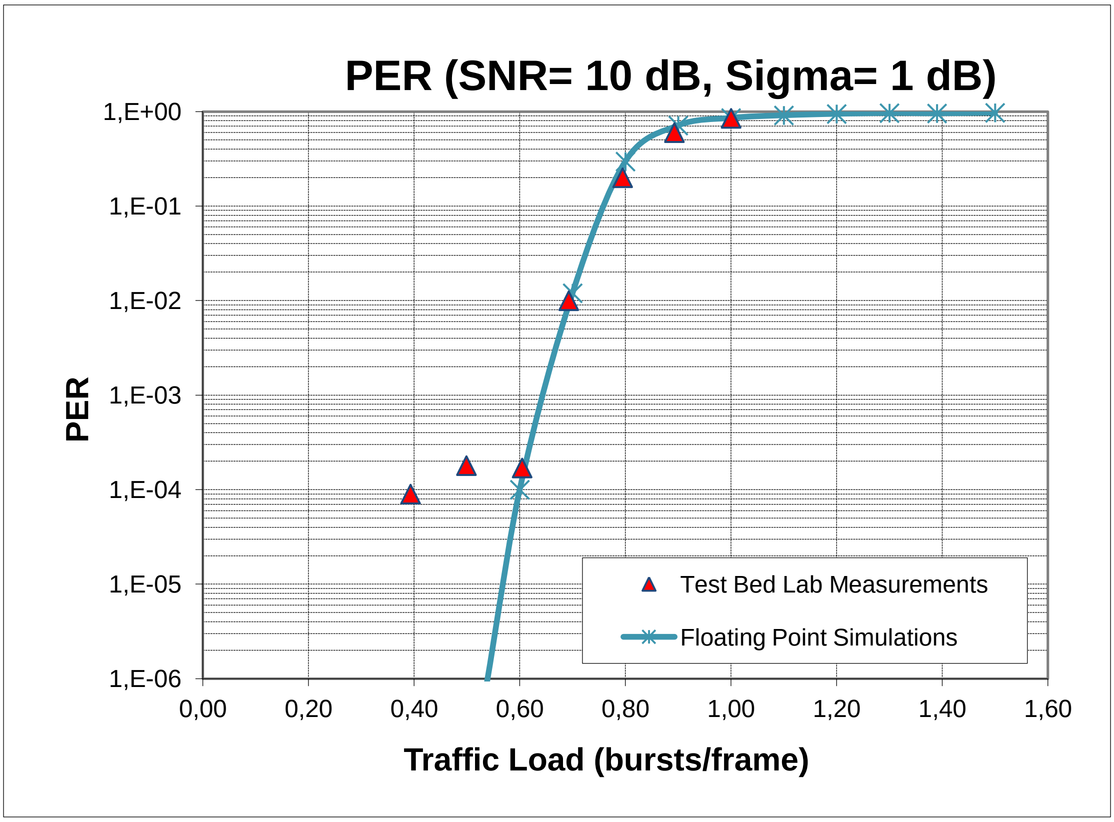


Figure B.10: Physical Layer Packet Error Rate

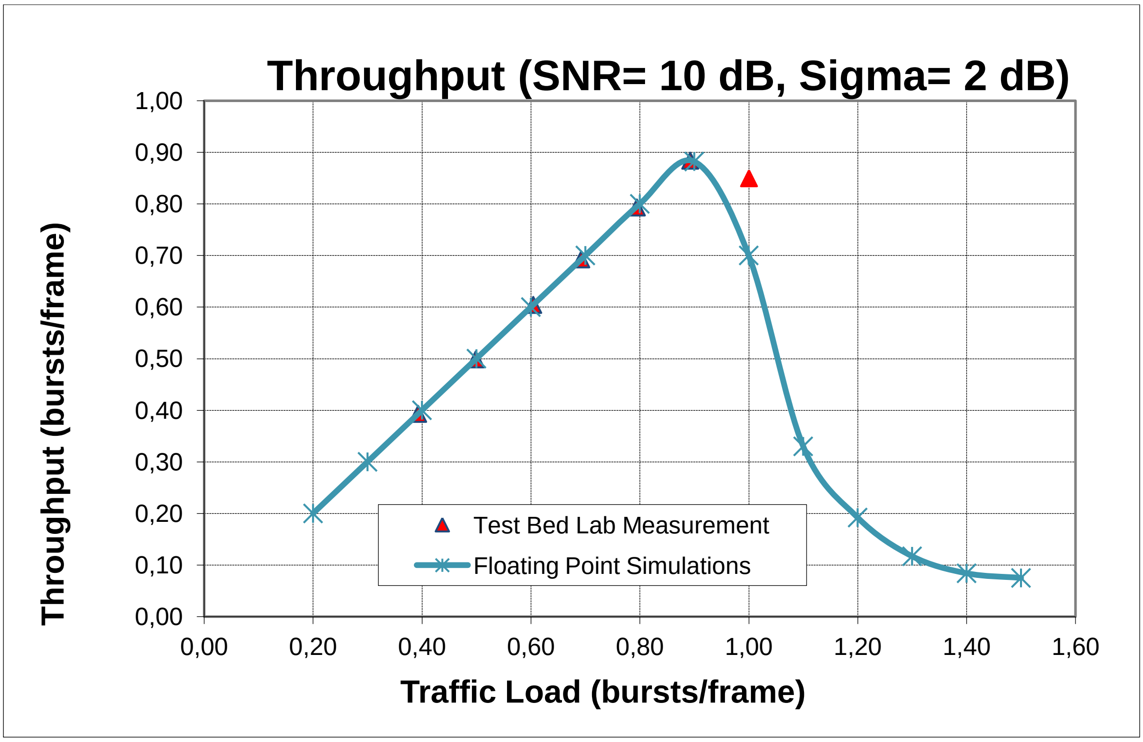


Figure B.11: Physical Layer Throughput

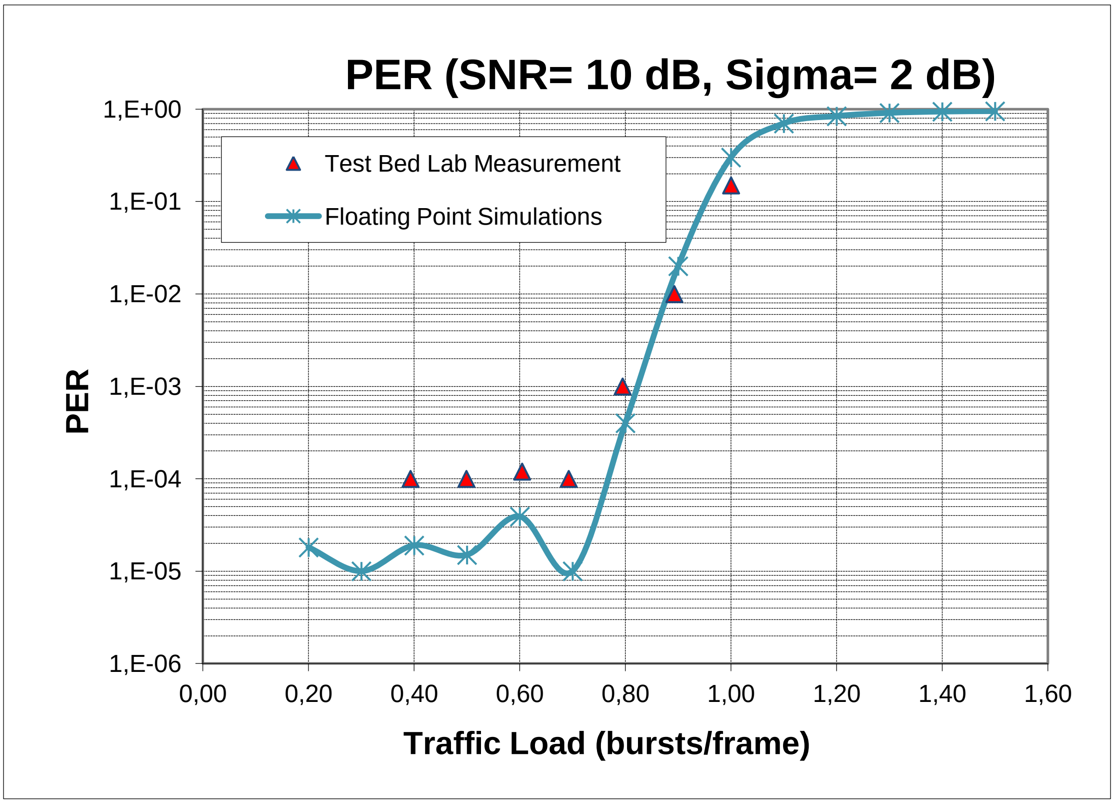


Figure B.12: Physical Layer Packet Error Rate

Annex C: Evolution towards SC-FDMA

This annex describes the possible merits of using the Single Carrier Frequency Division Multiple Access (SC-FDMA) scheme in a future evolution of the DVB-RCS2 standard. The SC-FDMA scheme has been adopted for use in the up-link of LTE [i.64] and, more recently, in the satellite component of DVB-NGH [i.65].

C.1 Introduction

Potential advantages of the SC-FDMA scheme in the DVB-RCS2 context are the possible increase in spectral efficiency due to the possible use of zero roll-off waveforms as well as simplification in the implementation of Multi Carrier Demodulators (MCD) at the GW or at the terminal side in mesh applications. It is also worth to note that an SC-FDMA receiver chain would have significant similarity with classic OFDM receiver chains which are commonly used in terrestrial applications.

The SC-FDMA scheme has also other important advantages. In particular, the most relevant one is the powerful equalization capability which allows to effectively counteracting frequency selective fading. This property is not relevant in the DVB-RCS2 context and will not be further considered here.

Conversely, the main disadvantage of SC-FDMA is the requirement for synchronization between users and the associated need of a Cyclic Prefix (CP) to absorb residual timing errors between users. The CP was introduced in OFDM/OFDMA like schemes to counteract the channel impulse response dispersion thus allowing ISI free operation. In the present context the channel is assumed non-dispersive. However, CP can be used to absorb asynchronicity between transmitting users. If timing error between users is within the CP, a single FFT can be used at the receiver for demultiplexing all different users thus significantly simplifying the MCD hardware implementation.

Regarding user timing synchronization accuracy requirements, this is not necessarily more stringent than in current RCS2 systems. In fact, actual acceptable timing errors are depending on the selected system parameters, i.e. FFT size, frequency resolution and CP overhead as will be discussed later in this annex.

The basic access scheme proposed here for SC-FDMA is still MF-TDMA with basically the same organization as in conventional DVB-RCS(2) systems. In particular, the MF-TDMA frame can be organized in time frequency slots as in current RCS system, with most of the slots used for carrying traffic and some others for control purposes (i.e. for signalling and synchronization acquisition/ maintenance).

C.2 SC-FDMA Principles

SC-FDMA differs from straight OFDMA for its use of DFT precoding for reducing envelope fluctuation of the generated signal (Figure C.1). Different strategies for sub-carrier mapping can be devised [i.66]. Two of the possible mapping strategies are indicated in Figure C.2: Localized FDMA (LFDMA) and Interleaved FDMA (IFDMA). In the former strategy each user is allocated a set of contiguous tones. With IFDMA tones allocated to a user are interleaved with those of the other users. For higher flexibility in resource management, the tone mapping strategy could be deferred at the time of actual resource allocation without an a-priori mapping strategy being defined.

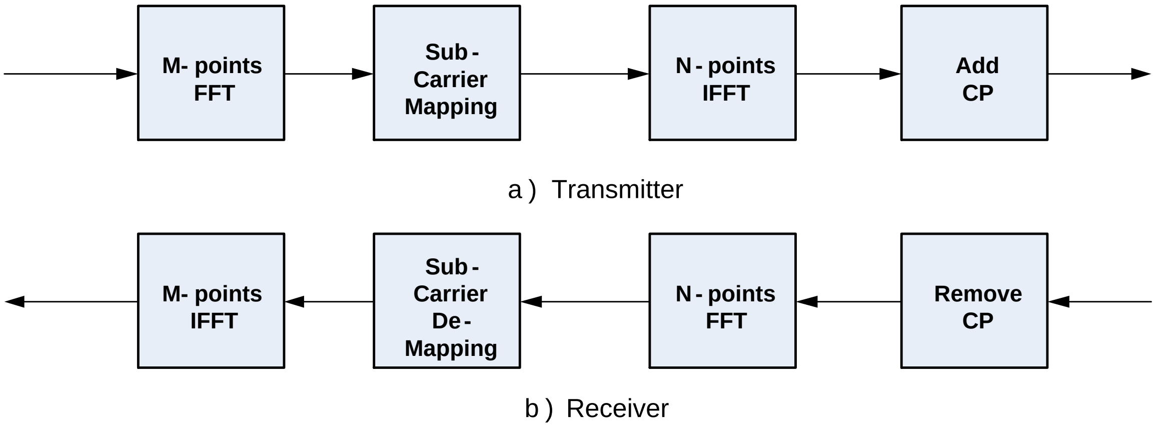


Figure C.1: SC-FDMA Schematic Diagram. It is assumed that M < N

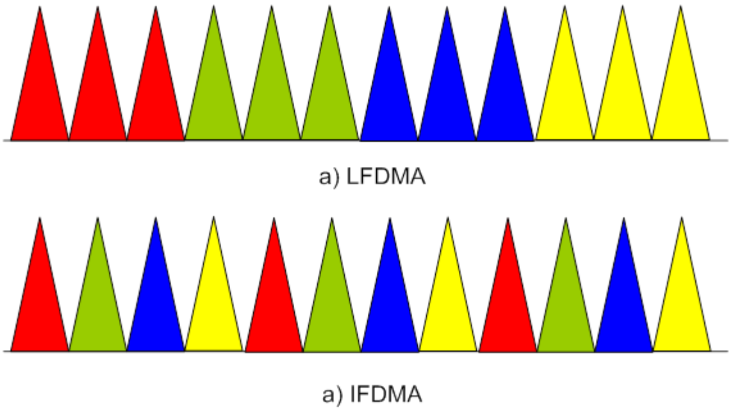


Figure C.2: Two common sub-carrier mappings in SC-FDMA

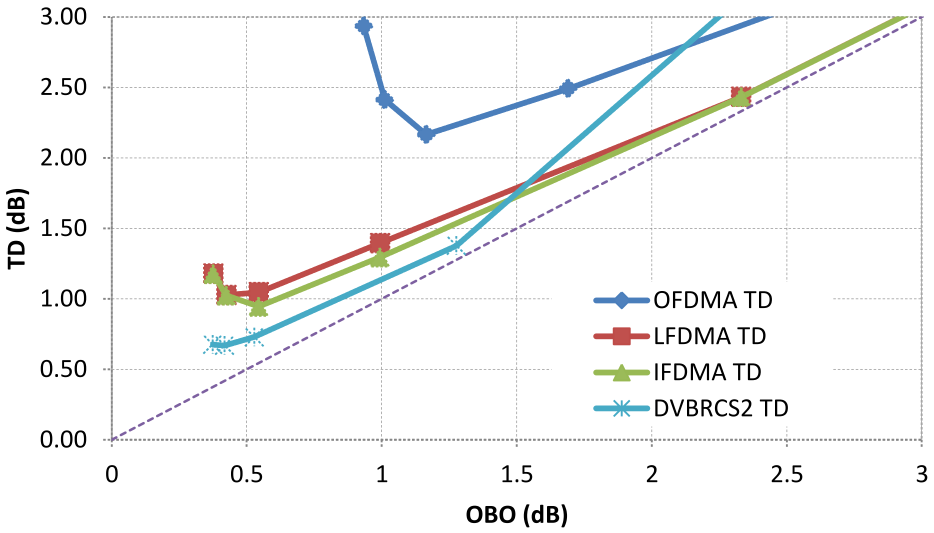
It should be noted that the LFDMA approach is equivalent to perform a sin(x)/x interpolation of the original time domain symbols. The obtained signal is actually the same as that obtainable by a time domain generation with ideal Nyquist pulse shaping filter (zero roll-off shaping filter). The use of zero roll-off implies some higher signal envelope fluctuation of the generated signal with respect to that of signals with higher roll-off. Such envelope fluctuation are however, still much lower than that experienced in true multicarrier (OFDM) systems. It should also be noted that, with SC-FDMA a non-zero roll-off can also be implemented still operating in the frequency domain. With the same roll-off, TDMA and SC-FDMA have the same PAPR. However, a zero roll-off is most often used in SC-FDMA, for example in [i.64] and [i.65], given its bandwidth advantage.

The IFDMA subcarrier mapping produces a repetition of the corresponding time domain symbols apart for a phase rotation applied between successive copies of each input symbol. No envelope signal fluctuations are thus generated (at least in the ideal case).

In practice, PAPR advantages of IFDMA mapping over LFDMA are minimal when the need for signal oversampling is considered and LFDMA is actually preferred to IFDMA for the lower sensitivity to synchronization errors.

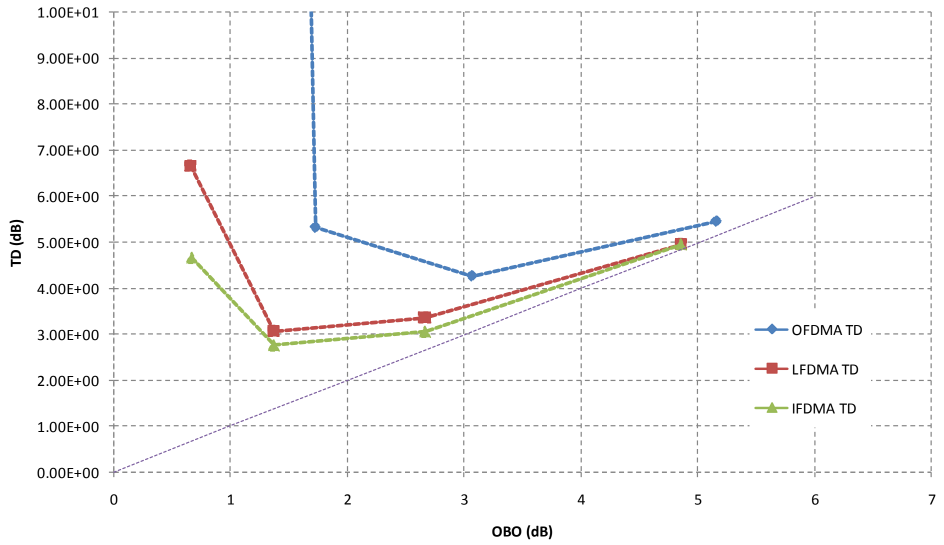
Figure C.3 shows the Total Degradation (TD) experienced by a terminal transmitting multiple carriers (8 carriers per terminal) assuming to use straight OFDMA transmission (i.e. with no DFT precoding) or SC-FDMA (either LFDMA or IFDMA). The advantages of SC-FDMA with respect to OFDMA are evident from the figure. Also the losses with respect to DVB-RCS2 are quite modest (about 0,4 dB). For completeness, Figure C.4 also shows the resulting TD assuming 16QAM modulation.

We stress that in case a single tone is assigned to a user, the DFT precoding becomes a no-op and the transmitted signal is actually equivalent to a signal with rectangular pulses. In such case no envelope fluctuation is present and no degradation from non-linearity would be experienced as with CPM modulation.



NOTE: A linearized TWTA was used at the terminal (see annex H of DVB-S2 specifications [i.2]).

Figure C.3: QPSK ½ total degradation (TD) curves evaluated at BER ≈ 10^{-3} (15 users each transmitting 8 tones)



NOTE: A linearized TWTA was used at the terminal.

Figure C.4: 16QAM ½ total degradation (TD) curves evaluated at BER ≈ 10^{-3} (15 users each transmitting 8 tones)

C.2a Demodulator Processing

We assume here that user terminals have already achieved a coarse timing (and frequency) synchronization before transmitting traffic bursts. As for conventional RCS terminals it is assumed that receiver synchronization is first achieved before transmitting the first burst on a random access slot.

Before going into steady state, terminals have to achieve a time synchronization accuracy such that the residual error is within the adopted CP (as briefly discussed in clause C.5). Assuming that terminals are already in steady-state synchronization and transmit traffic burst and control burst for synchronization /signalling purposes (which we here refer to as SYNC slot in analogy with first generation RCS terminology) on the NCC assigned resources.

The demodulator processing, which we briefly illustrate below, can be used for both Traffic and Control bursts.

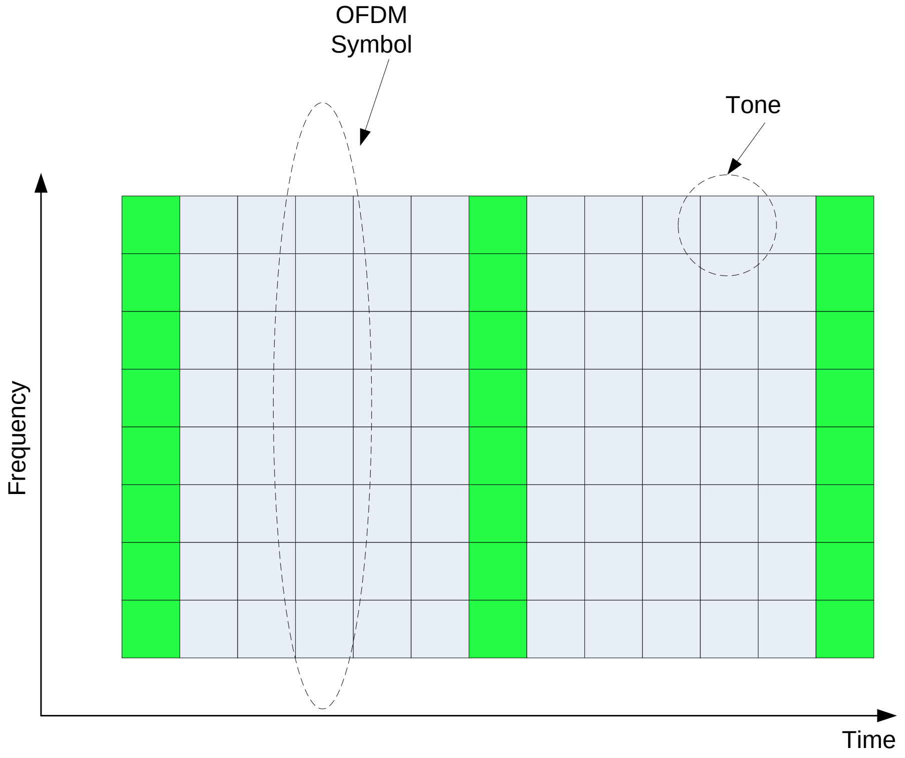
The following definitions are used for "modulation symbols", "OFDM symbols" and "tones":

- **modulation symbols**, the sequence of QPSK or 8PSK or 16QAM symbols as per a conventional TDMA system.
- **OFDM symbols:** indicate the signal obtained after DFT precoding of the "modulation symbols. Hence, an OFDM symbol will include from a minimum of 1 "modulation symbol" to a number of "modulation symbols which is only limited by the system bandwidth (and frequency granularity of SC-FDMA).
- **Tones:** In order to avoid confusion between "OFDM symbols" and "modulation symbols", we will refer to the "modulation symbols" as tones given the fact that with DFT precoding a number M of "modulation symbols" is transformed into an equal number of tones in the frequency domain (although there is no one-to one correspondence between a modulation symbol and a tone).

A burst is assumed composed of a sequence of encoded payload OFDM symbols intermixed with pilot OFDM symbols. Pilot OFDM symbols are added with a certain constant periodicity. The burst is assumed starting with a pilot OFDM symbol. The final OFDM symbol of a burst is also assumed to be a pilot OFDM symbol. This last pilot OFDM symbol may be at a lower distance from the previous pilot OFDM symbol, due to the fact that the encoded payload length in OFDM symbols may not be a multiple of the chosen pilot OFDM symbol repetition period.

By pilot OFDM symbol repetition period, we intend the number of payload OFDM symbols comprised between two pilot OFDM symbols (the actual number may be lower when considering the last pilot OFDM symbol of a burst).

Burst organization is pictorially shown in Figure C.5.



NOTE: An OFDM symbol is here composed of either all pilot tones (green one) or payload symbols.

Figure C.5: Organization of a SC-FDMA burst after DFT precoding. Green Box indicates pilot tones

The reason to have a given OFDM symbol composed of either all pilot tones or all information symbols is due to the fact it is desirable to have constant amplitude pilot tones in the frequency domain (hence, not only in the time domain) for reasons connected to channel estimation. Channel estimation is, in fact, easier to perform in the frequency domain. At this regard time error estimation requires multiple pilot tones to be present in a single OFDM symbol.

The overall sequence of operation of the Demodulator is shown in Figure C.6. The demodulator, in addition to demodulated data, also provides measurement which may be fed-back to transmitter for adjusting their parameters (time, frequency, power level, possible MODCOD).

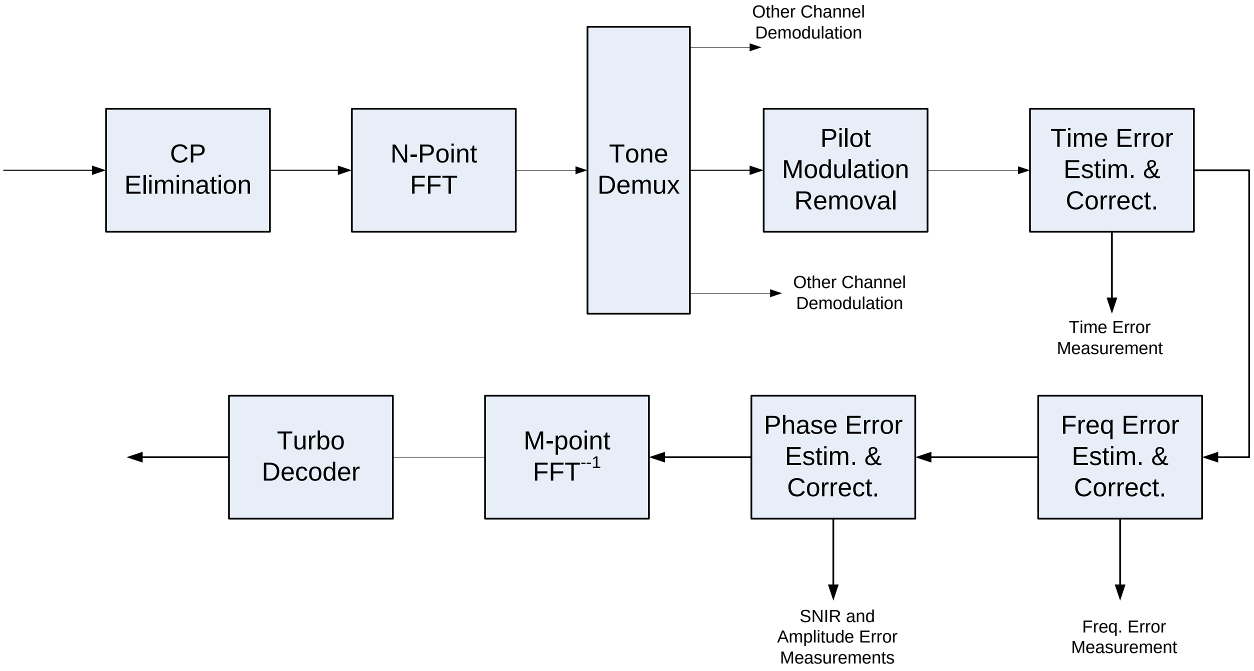

CP Elimination
N-Point FFT
Tone Demux
Other Channel Demodulation
Pilot Modulation Removal
Time Error Estim. & Correct.
Other Channel Demodulation
Time Error Measurement
Turbo Decoder
M-point FFT--1
Phase Error Estim. & Correct.
Freq Error Estim. & Correct.
SNIR and Amplitude Error Measurements
Freq. Error Measurement

Figure C.6: Schematic Diagram of the demodulator

It is generally convenient to have constant tone amplitude in both the frequency domain and time domain. This has led to the widespread usage of the Zadoff-Chu sequences (also known as a CAZAC sequence - constant amplitude zero autocorrelation waveform) as pilot symbols in SC-FDMA.

The use of such sequences is assumed here for pilot symbols although in applications where the channel is not frequency selective, having constant amplitude of pilot tones in the frequency domain is not strictly required. Concentrating pilot energy at the edge of the band may be, in fact, a better strategy as it would allow higher accuracy in timing error measurements. Also, to reduce the pilot overhead, mixed pilot and traffic symbols are often used. This leads to higher envelope fluctuation on those mixed symbols. The associated performance penalty is however acceptable given that such higher envelope fluctuation is concentrated in a small percentage of the symbols. Moreover, some solutions for controlling the PAPR of the mixed symbol when employing SC-FDMA already exist (e.g. pilot design of the DVB-NGH satellite component [i.65]). Performance results presented here do not assume the use of mixed pilot symbols and make use of conventional Zadoff-Chu sequences for such pilot symbols.

Looking at the scheme in Figure C.6 it appears that after CP elimination and transformation of the signal in the frequency domain (through the N-point FFT), tone demux can be performed in order to separate tones according to their users. Once tones are separated, narrowband processing per user can take place. All subsequent demodulation algorithms can be performed in the frequency domain, coming back to the time domain (with the M-point IFFT) only for final decoding of the data.

The symbol timing recovery can be carried out in the frequency domain thanks to the duality of time and frequency with respect to Fourier Transform (a time shift in the time domain is equivalent to a linear phase ramp in the frequency domain). Hence, estimation of timing error in the frequency domain can actually be performed with conventional algorithm for time-domain frequency estimation.

For the actual timing error estimator the Mengali-Morelli frequency estimator [i.67] has been used to get the performance results which will be presented in clause C.3.

Once the timing error has been estimated, timing correction is performed by rotating the phase of each tone according to the corresponding frequency-domain caused linear phase drift per tone within each OFDM symbol.

Frequency Offset Recovery also operates on the pilot tones. However, processing is now done between pilot tones of the same frequency index but on different pilot OFDM symbols.

After frequency error estimation, frequency correction takes place by rotating the phase of tones in subsequent OFDM symbols.

It should be noted that an implicit assumption here is that the frequency error is always less than a frequency bin. In this case there is no need of coarse frequency estimation as the frequency error is always less than a frequency bin. This is certainly a reasonable assumption for terminals already in steady-state synchronization status.

C.3 Performance Results

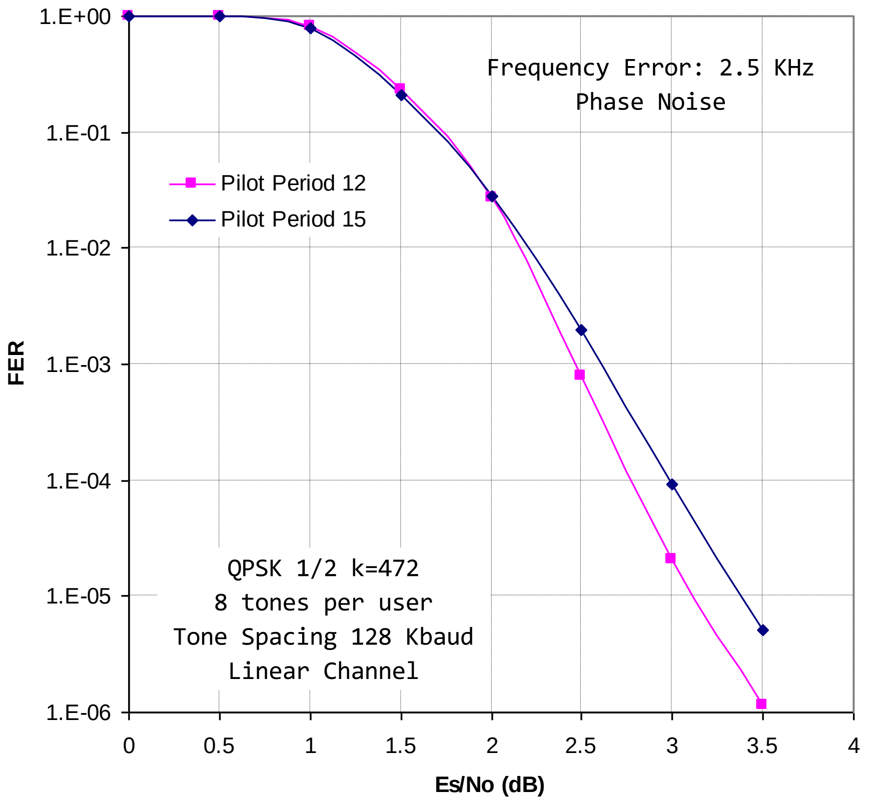
Some simulation performance results are reported here for both QPSK r = ½ and 8PSK r = 2/3 with a Turbo Coding scheme in line with the RCS2 normative document [i.1].

Regarding SC-FDMA simulation parameters, a tone granularity of 128 KHz is assumed. Two adjacent users were simulated each with 8 tones (for a total baud rate per user of 1,024 MBaud) out of a total number of tones (size of the wideband FFT) equal to 128. A 128 point FFT was thus used at the receiver side to demultiplex the various users which are then subject to independent demodulation.

Figure C.7 shows the performance in linear channel of QPSK 1/2. Payload size of the bursts was 472 info bits according to the waveform ID=4 of annex A of the normative document [i.1]. One of the SC-FDMA users had pilot periodicity equal to 12 OFDM symbols (corresponding to a total of 7 OFDM symbols used for pilots within the burst) and the other had pilot repetition equal to 15 OFDM symbols (corresponding to a total of 5 OFDM symbols used for pilots within the burst). Frequency error was 2,56 KHz for both users. Phase noise according to DVB-RCS guideline was added. Timing error for both users was different from zero but lower than the Cyclic Prefix (CP).

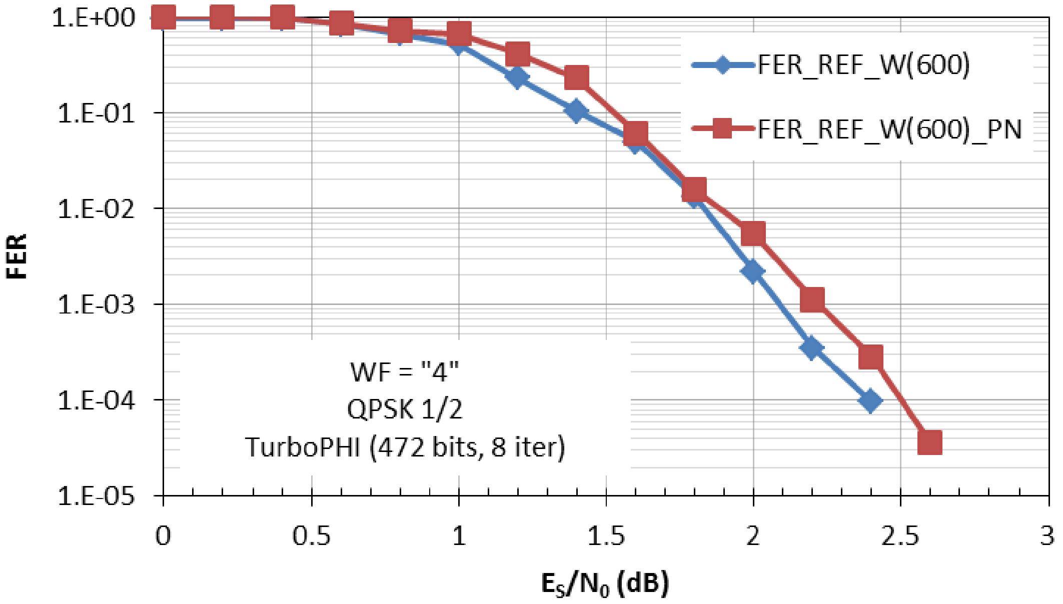
The corresponding reference performances for the DVB-RCS2 waveform are shown in Figure C.8. The performance loss of the SC-FDMA case (with pilot periodicity equal to 12 OFDM symbols) is about 0,3 dB.

Performance of the SC-FDMA system in presence of a non-linearity (modelled according to the Rapp model [i.38]) is shown in Figures C.9 and C.10 for pilot periodicity of 15 OFDM symbols and 12 OFDM symbols, respectively. The AM/AM of the non-linearity is shown in Figure C.11. The loss with respect to the standard ID=4 waveform of DVB-RCS2 is only 0,35 dB for the case of pilot periodicity 12 at IBO = 0 dB. To such value about 0,1 dB should be also added for the difference in OBO due to the use of a zero roll-off factor in SC-FDMA.



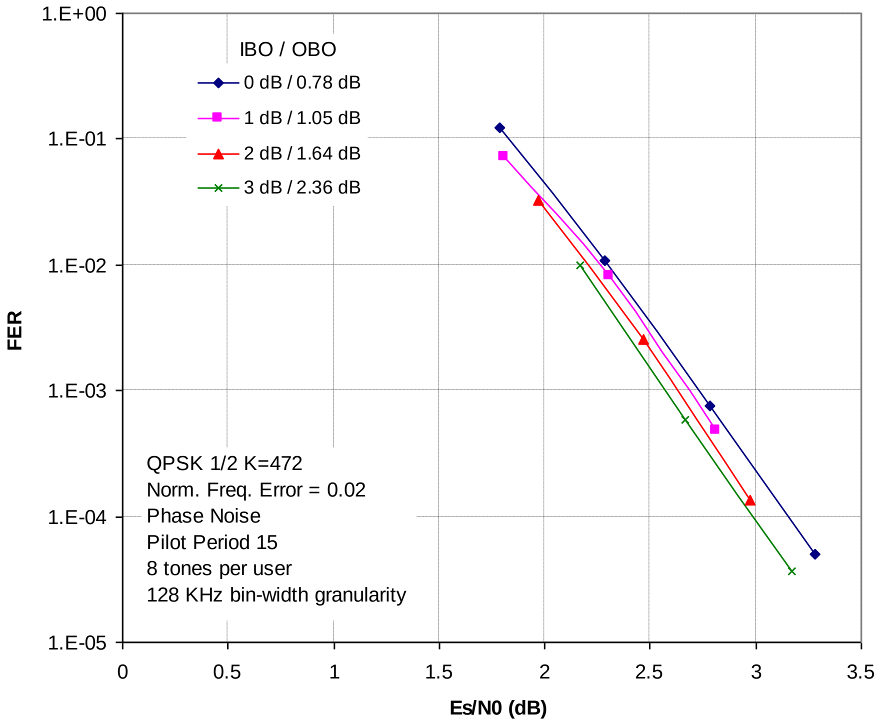
NOTE: Payload size 472 info bits.

Figure C.7: Performance in linear channel of SC-FDMA QPSK 1/2



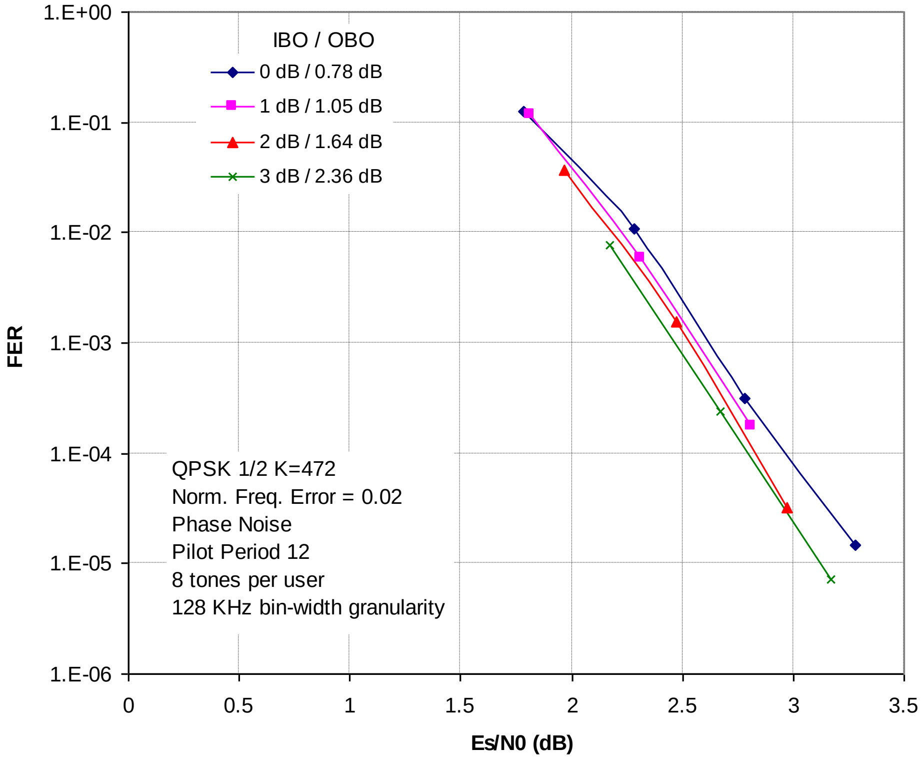
NOTE: Payload size 472 info bits.

Figure C.8: Reference performance of DVB-RCS2 (Waveform ID=4) in linear channel with and without phase noise



NOTE: Payload size 472 info bits.

Figure C.9: Performance of SC-FDMA with QPSK ½ (pilot period 15) in a nonlinear channel with different IBO



NOTE: A RAPP model HPA was considered. Payload size 472 info bits.

Figure C.10: Performance of SC-FDMA with QPSK ½ (pilot period 12) in a non-linear channel with different IBO

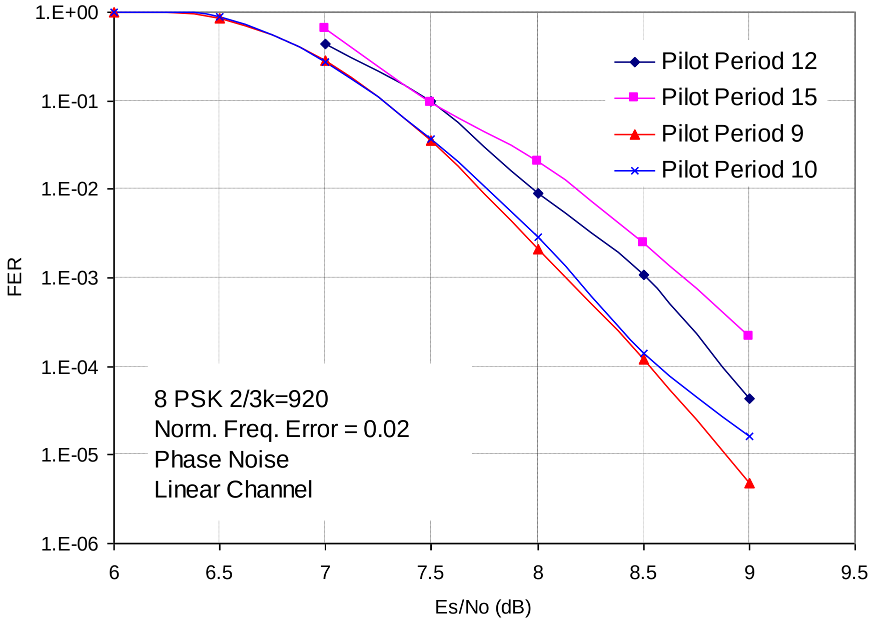
As a second study case, a payload of 920 info bit with 8 PSK modulation and r = 2/3 was considered (Waveform ID=8 of annex A in the normative document [i.1]). Performances with different tone periodicity are shown in Figures C.11 and C.12 respectively for a linear and non-linear channel. Two adjacent users were simulated each with 8 tones (tone granularity 128 KHz). Frequency error was about 2,56 KHz for both users. Phase noise according to DVB-RCS guideline was added.

Using the same simulation models, the performance of conventional waveform ID=8 of [i.1] has been obtained as shown in Figures C.13 and C.14 respectively for the linear and non-linear case.

For the non-linear channel (with IBO = 0 dB) SC-FDMA (with pilot periodicity equal to 9 OFDM symbols) has a loss of about 0,3 dB (0,4 dB when the OBO difference is also accounted) with respect to the reference DVB-RCS2 case.

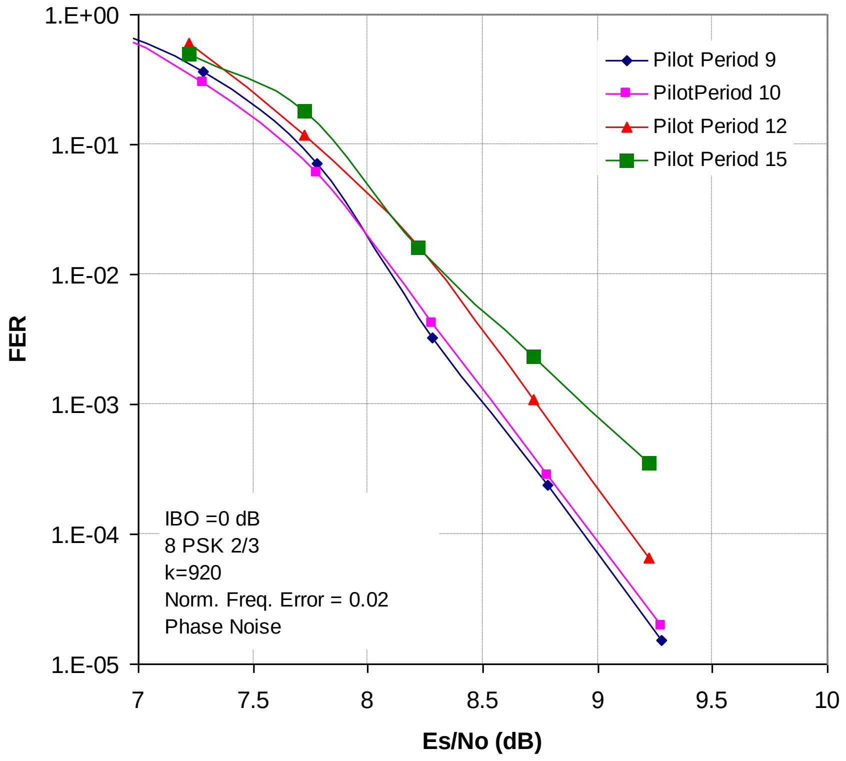
The small difference in performance between SC-FDMA and DVB-RCS2 is mostly due to the frequency error impact. For smaller frequency error the performance difference is smaller approaching the difference in OBO. The impact of the frequency offset can be reduced by using mixed data and pilot symbols, which would allow having more frequent pilots in the time domain without increasing the overall pilot overhead.

The small performance loss of SC-FDMA is anyway compensated by the higher spectral efficiency achievable by SC-FDMA.



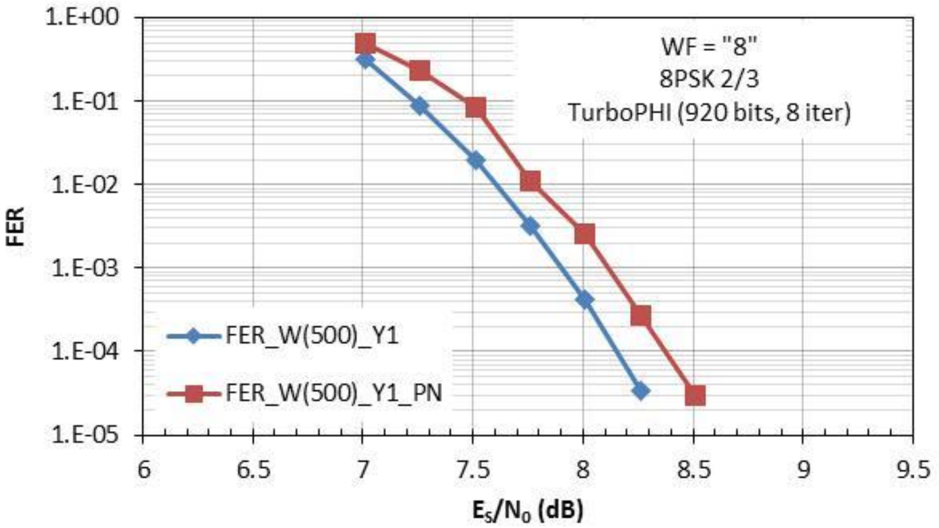
NOTE: Payload size 920 info bits.

Figure C.11: Performance in linear channel of SC-FDMA 8PSK 2/3



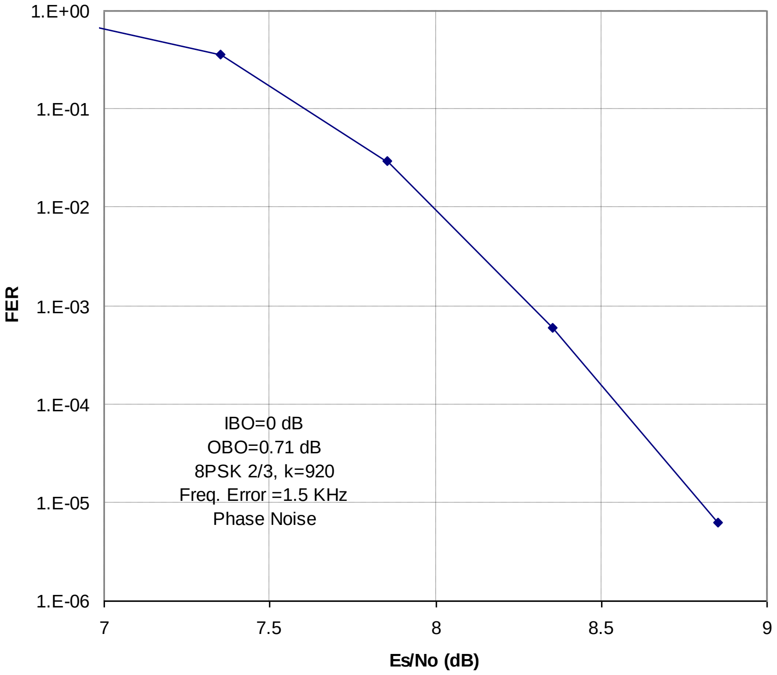
NOTE: Payload size 920 info bits.

Figure C.12: Performance in non-linear channel of SC-FDMA 8PSK 2/3 with IBO = 0 dB (OBO = 0,78 dB)



NOTE: Payload size 920 info bits. Waveform ID=8.

Figure C.13: Performance in linear channel of standard DVB-RCS2 8PSK 2/3 (with and without phase noise)



NOTE: Payload size 920 info bits. Waveform ID=8.

Figure C.14: Performance in non-linear channel of DVB-RCS2 8PSK 2/3 at IBO = 0 dB

Tables C.1 and C.2 show spectral efficiencies respectively achieved in DVB-RCS2 with waveform ID=4 and 8 and SC-FDMA with QPSK ½ (pilot periodicity 12 OFDM symbols) and 8PSK 2/3 (pilot periodicity 9 OFDM symbols) for two different CP lengths.

It can be observed that for both waveform examples, SC-FDMA provides a spectral efficiency improvement with respect to RCS2 with a small performance loss (less than 0,4 dB at FER = 10^{-4}). It should however be noted that the current simulation results are based on pilot distribution which was not optimal as the last OFDM pilot symbols was not spaced as the other symbols. The spectral efficiency of SC-FDMA could be likely further increased by considering mixed data and pilot symbols and better payload length. In DVB-NGH for example, for the SC-FDMA satellite component, mixed data and pilot symbols with controlled low PAPR are used [i.65].

Table C.1: Overall Spectral efficiencies of two DVB-RCS2 standard waveforms

DVB-RCS2 Waveform	Useful Bits	Payload Symbols	Preamble / pilots	Guard symbols	Roll-Off	Spectral Efficiency bit/s/Hz
ID=4 (QPSK ½)	472	472	64	6	0,2	0,738
ID=8 (8PSK 2/3	920	460	76	6	0,2	1,447

Table C.2: Overall Spectral efficiencies of two SC-FDMA mode roughly having the same power efficiencies as the corresponding DVB-RCS2 ones of previous table.

SC-FDMA Waveform	Useful Bits	OFDM Payload Symbols	OFDM Pilot Symbols	Spectral Efficiency (bit/s/Hz)	
				CP=1/8	CP=1/16
QPSK ½	472	59	7	0,795	0,841
8PSK 2/3	920	57,5	9	1,57	1,67

C.4 Implementation Aspects

The SC-FDMA scheme has other attractive properties as mentioned in the introduction. With regard to Multi-Carrier Demodulator implementation, SC-FDMA is advantageous compared to conventional techniques when implementing MCD capability at the terminal as demultiplexing of different user signals is straightforward via FFT processing.

As an example, a single 512-points FFT processor at the mesh receiver demodulator would be able to extract any signal within a 36 MHz transponder with a frequency tone resolution of around 80 KHz allowing compatibility with any allocated user bandwidth which is an integer multiple of 80 KHz. The 512-point FFT would require 3 204 real multiplication and 12 420 real additions should a radix-8 algorithm be selected. This is equivalent to about 6 real multiplications and 25 real additions per input complex sample. Should a real sampling be used, a 1 024 points FFT operating on a signal sampled at about 80 MHz is required. However such 1 024 FFT can still be computed via a complex 512-points FFT provided that an additional stage performing the real-to-complex conversion is added to the FFT processor (Figure C.15). The last stage for real to complex conversion will require 1 018 real multiplications and 2 560 real additions. Even with this addition we have about 8 multiplication and 30 additions per complex sample. At 40 MHz complex sampling rate these would amount to 320 Million Multiplies/s and 1,2 Billion additions/s.

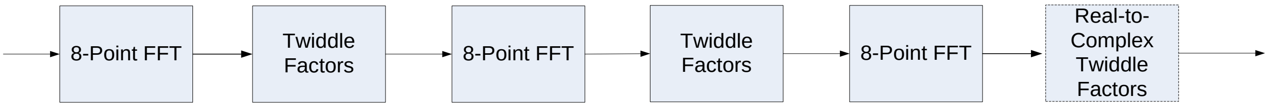


Figure C.15: A 512-point complex FFT with radix-8 decomposition. Adding a stage for real to complex conversion, a 1 024 real FFT can be computed (see dashed block in the figure)

Should the same capability be implemented in a conventional system, it would have required to implement for example a fast convolution approach (Figure C.16) with a fixed size FFT (in the order of at least 8K points if we want to minimize guard band between channels) followed by spectral modification and a number of smaller size IFFT (one per actual carrier to be separated). Overlapped processing of the windows is also required. The Direct FFT is common for all signals to be demultiplexed. A separate IFFT (and inverse windowing if windowing is used) should be done for each carrier.

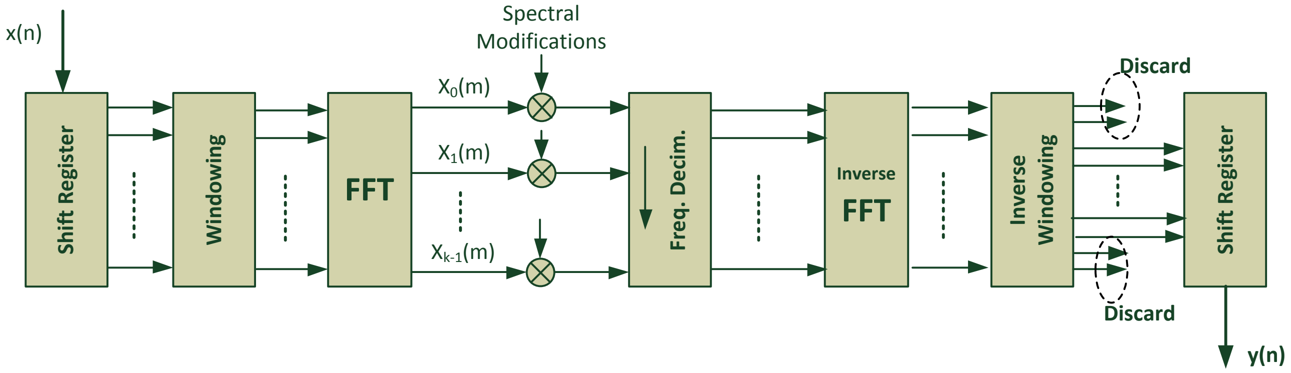


Figure C.16: Demultiplexer implemented via Fast_Convolution

The direct FFT, if implemented with a radix 2 algorithm, requires about 200 000 real multiplications or 24 real multipliers per input sample. Assuming an overlapping factor between FFTs of 25 %, the direct FFT would become equivalent to about 30 real multipliers /sample. To that number, the complexity of the IFFTs has to be added. The actual impact of IFFTs would depend on the actual usage of the channel. Assuming IFFTs complexity is equal to that of the FFT we would end up with 60 real multiplications/sample just for the FFT/IFFTs, i.e. 10 times more than with a 512 point FFT.

The complexity ratio of the two demultiplexing approaches is at least of a factor 10 in terms of number of multiplications in favour of the SC-FDMA approach. Hence, SC-FDMA would actually provide a cost saving in the modem at least when multi-carrier reception capability is desired.

Another potential advantage of SC-FDMA is its flexibility in resource assignment. In conventional MF-TDMA, carrier bandwidths have to be defined in advance according to the foreseen needs of the system and may only be reconfigured occasionally. So a few carrier bandwidths are typically available and this would make interference and fading mitigation techniques like DRA (Data Rate Adaptation) not very effective. In SC-FDMA, instead, resources can be dynamically assigned in quantum of both time and frequency. DRA would become a quite natural way to overcome fading in such a case. We recall at this regard that DRA, when feasible, is more efficient than ACM as fading countermeasure.

C.5 Synchronization Requirements

In order to exploit the SC-FDMA technology it is needed that user terminals are mutually time synchronized with a maximum error less than the CP width.

The absolute time accuracy which is required with SC-FDMA is thus dependent on the CP duration.

For a given CP relative overhead, the absolute time duration of the CP can be made larger by increasing the duration of the OFDM symbol. For a given bandwidth, the duration of the OFDM symbols is increased by increasing the FFT size (hence reducing the frequency bin width).

As examples, for a 1/8 CP size and about 41 MHz total gross bandwidth spanned by the FFT, Table 3 shows the duration of the CP as function of the FFT size.

Table C.3: Relationship between CP duration and FFT parameters

FFT Size	Total bandwidth (MHz)	Frequency Bin Width (kHz)	OFDM Symbol duration (μs)	CP rel. Overhead	CP Duration (μs)
128	40,96	320	3,125	1/8	0,39063
256	40,96	160	6,25	1/8	0,78125
512	40,96	80	12,5	1/8	1,5625
1 024	40,96	40	25	1/8	3,125
NOTE: Duration of the OFDM symbol is without CP.					

It appears that timing accuracy requirements are similar or slightly more stringent than those of required in DVB-RCS2 system. It should be noted that longer FFTs might also require a corresponding increase of the frequency accuracy to reduce degradation due to user frequency errors.

Synchronization requirements of SC-FDMA are thus not necessarily more stringent than those of conventional MF-TDMA systems. The same frame organization principles as in current DVB-RCS may be applied to SC-FDMA.

Regarding random access bursts like logon burst which is typically transmitted before having the opportunity of achieving strict TX synchronization, several alternatives can be considered. In the preferred alternative, such slots are segregated in a reserved temporal section of the frame in order that their lacks of synchronization do not produce interference on frequency adjacent users. Random access bursts transmitted in such slots may then use a different waveform than SC-FDMA. These bursts can be used for initial logon and for acquiring steady-state synchronization.

C.6 Compatibility with normative specifications

The Current RCS2 Lower Layer normative document does not foresee the use of SC-FDMA. Transmission roll-off is fixed to 20 % for RCS2 Linear Modulation against the 0 % roll-off which should be preferred for SC-FDMA. Also, CP extent needs to be defined for SC-FDMA. Current signalling for the Data Block for the TC-LM Transmission Format Class (see Table 6.18 of the normative document [i.1]) is however sufficient for describing SC-FDMA bursts.

Annex D: Time Slot Sharing

D.1 Introduction

Time Slot Sharing (TSS) is an effective solution for increasing the bandwidth efficiency of mesh connections over transparent satellites. The technique relies on the fact that two peer terminals, if served by the same beam, may reuse the same time slot as each peer is able to cancel its own signal from the received signal thus allowing it to demodulate the unknown signals coming from the other terminal.

D.2 Applicability

The following advantages of TSS can be highlighted:

- TSS is compatible with different transmit waveforms (independent of the modulation and coding). Both linear modulations and CPM are compatible with TSS. Different MODCODs could also be used by the terminals sharing the same time/frequency slot. Similarly ACM operations can be also supported.

- TSS can be used in both bandwidth limited and power limited situations. Whilst its advantages are obvious in a bandwidth limited situation, it may have also advantage in a power limited situation as the saved bandwidth provided by this approach may be exploited for reducing the spectral efficiency of transmitted carriers which implies an increase of the achievable power efficiency. As a matter of fact if the available bandwidth doubles with such a technique we may, for example, use QPSK ½ instead of 8PSK 2/3. Doing so we have the same spectral efficiency as in the original system but we gain in power as QPSK 1/2 requires an Eb/N0 about 2,5 dB lower than 8PSK 2/3 for a codeword size of about 500 bits.

The following constraint should be considered for the technique applicability:

- TSS can only be used when the transmitting earth station is also able to receive its own transmitted signal. That generally precludes its general adoption in multi-beam systems apart for the cases where the two peer entities happen to be in the same beam.

- Although TSS can also be employed in star configurations, it is best suited for symmetric links such as transparent mesh profile where peer-to-peer links can be established using balanced carriers (in terms of power and bandwidth).

- The use of TSS could induce constraints in the resource assignment strategy. Since it operates on peer-to-peer connections, it is necessary that the same resources (in terms of MF-TDMA slots) be assigned to both parties involved in a peer-to-peer connection. In addition to more complex (and perhaps less efficient) resource management, the traffic should be balanced. Such constraint will somewhat reduce the actual gain of the technique with respect to the theoretical maximum.

- TSS is sensitive to channel non-linearity. For linear modulations, terminal non-linearity effects could be mitigated by using the SSPA output as the reference signal. The use of constant-envelope CPM signals is effective since the CPM waveform is insensitive to terminal non-linearity. Satellite non-linearity, given the multicarrier satellite operation, cannot be however fully compensated through a linear canceller.

Maximum theoretical gain is a factor of two in case of bandwidth limited systems, assuming no limiting constraint due to the increase of power spectral density at the satellite input.

It should be noted that TSS is an analogue form of a network coding scheme similar to those proposed in literature to reduce bandwidth requirements in regenerative satellite systems.

TSS can be extended to multicast multi-party conference (see Figure D.1), although the theoretical gain rapidly decreases with number of parties in the conference as in general (N-1) slots are required for an N-party conference. The resulting theoretical gain is thus N/(N-1) which is maximum with just two users.

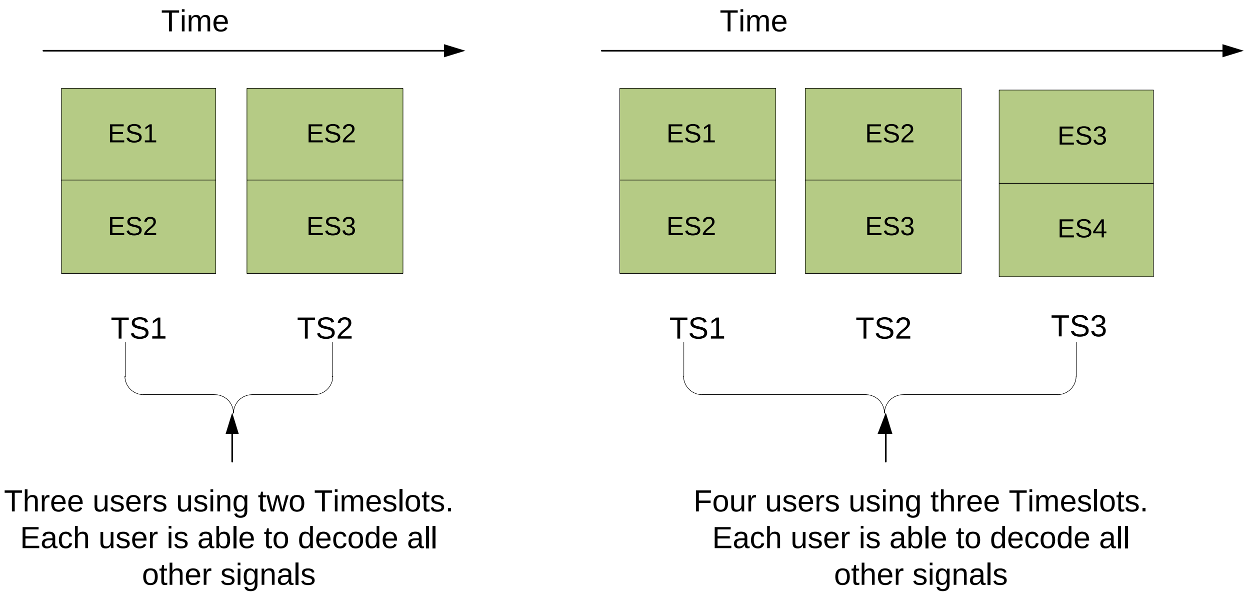

Time
ES1
ES2
ES2
ES3
TS1
TS2
Three users using two Timeslots. Each user is able to decode all other signals
Time
ES1
ES2
ES2
ES3
ES3
ES4
TS1
TS2
TS3
Four users using three Timeslots. Each user is able to decode all other signals

Figure D.1: TSS in Multi-Party Conference

D.3 TSS Echo Cancellation

The TSS technique relies on a peer terminal to cancel its own (round trip delayed) signal from the received signal. As shown in Figure D.2, an "echo" canceller is required at the receiver side to remove the self-interfering signal from the received signal. After Echo Cancellation a standard RCS2 demodulator can be used.

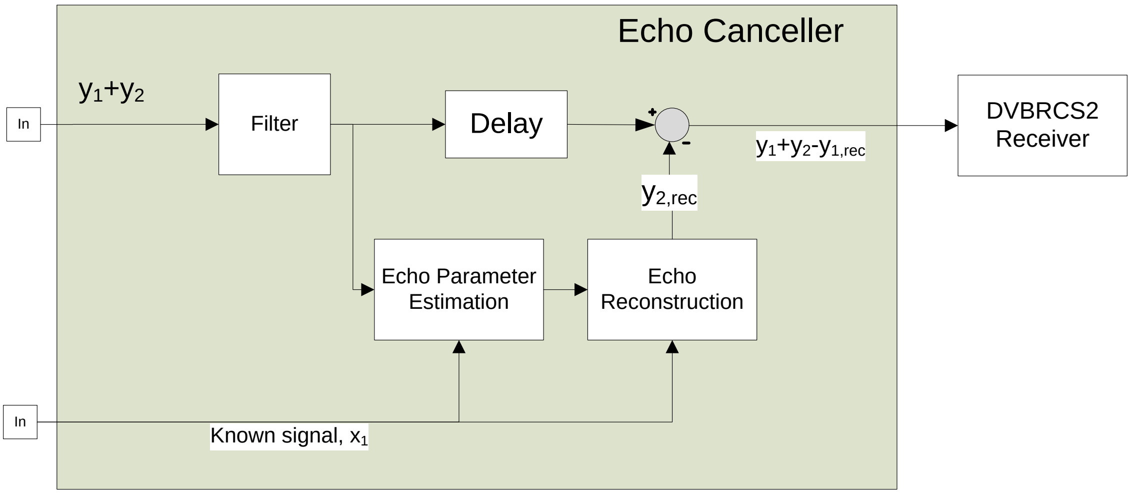


Figure D.2: TSS cancellation technique

The filter in the echo canceller may be a Digital Anti-Aliasing Filter (DAAF) to reduce noise outside the signal bandwidth. For linear modulation the DAAF may actually implement the shaping filter moving this function outside the standard DVB-RCS2 receiver.

The filtering, echo parameter estimation and echo reconstruction can be actually performed via a sequence of functions as identified in Figure D.3. These functions are independent of the modulation of the echo as the echo signal is fully known apart for parameters like frequency, phase and timing.

The only assumption here is that the known samples are provided with the same sampling rate as the actual sampling rate of the incoming signal to be processed. Such sampling rate it is assumed to be at least three times larger than actual signal bandwidth.

The minimum sampling frequency needs to satisfy Nyquist sampling theorem. However, a somewhat larger sampling frequency may be desirable with Farrow interpolation and to simplify fine timing recovery.

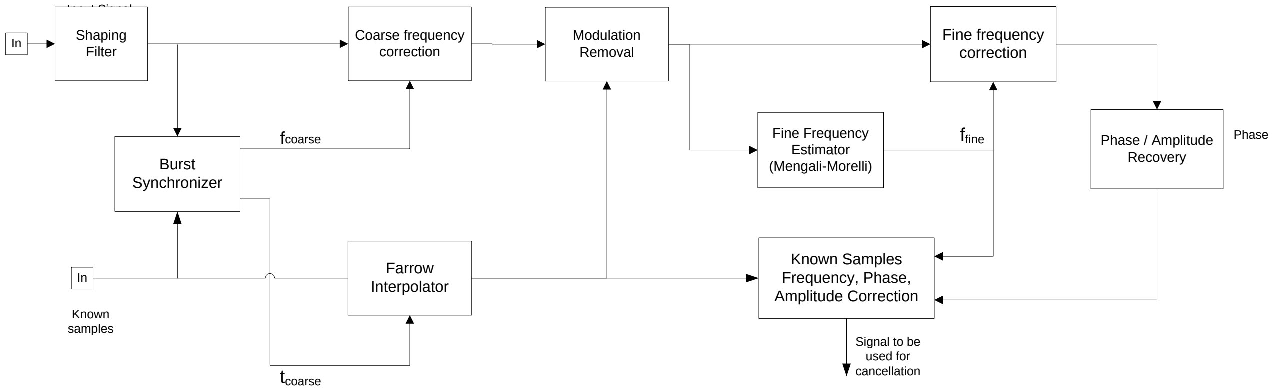

In
Shaping Filter
Coarse frequency correction
Modulation Removal
Fine frequency correction
Burst Synchronizer
f_{coarse}
Fine Frequency Estimator (Mengali-Morelli)
f_{fine}
Phase / Amplitude Recovery
Phase
In
Known samples
Farrow Interpolator
Known Samples Frequency, Phase, Amplitude Correction
Signal to be used for cancellation
t_{coarse}

Figure D.3: Echo parameter estimation and reconstruction

The first active process for the echo estimation and reconstruction is the burst synchronizer. This block performs simultaneously both the estimation of the arrival time of the echo and the coarse estimation of the echo frequency.

Given the unknown frequency of the echo, the matched filter needs to be split in smaller sections which are coherently combined according to different frequency error hypothesis (Figure D.4). The largest result for correlation in a time window corresponding to the time uncertainty range of the echo arrival is then selected as the best coarse timing and frequency hypotheses.

A fractional delay can also be estimated with respect to the echo Start of Burst sample by parabolic interpolation between the best correlation value achieved by the burst Synchronizer and the two adjacent correlation values.

A Farrow Interpolator is then used to compensate for such fractional delay. The input sequence of samples is then multiplied by the complex conjugate of the known echo interpolated samples to remove phase modulation thus obtaining a signal usable for fine frequency, phase and amplitude estimation.

The obtained frequency correction values and amplitude / phase values can then be also applied to the reference signal in order to have the echo replica ready for cancellation from the original input signal stream.

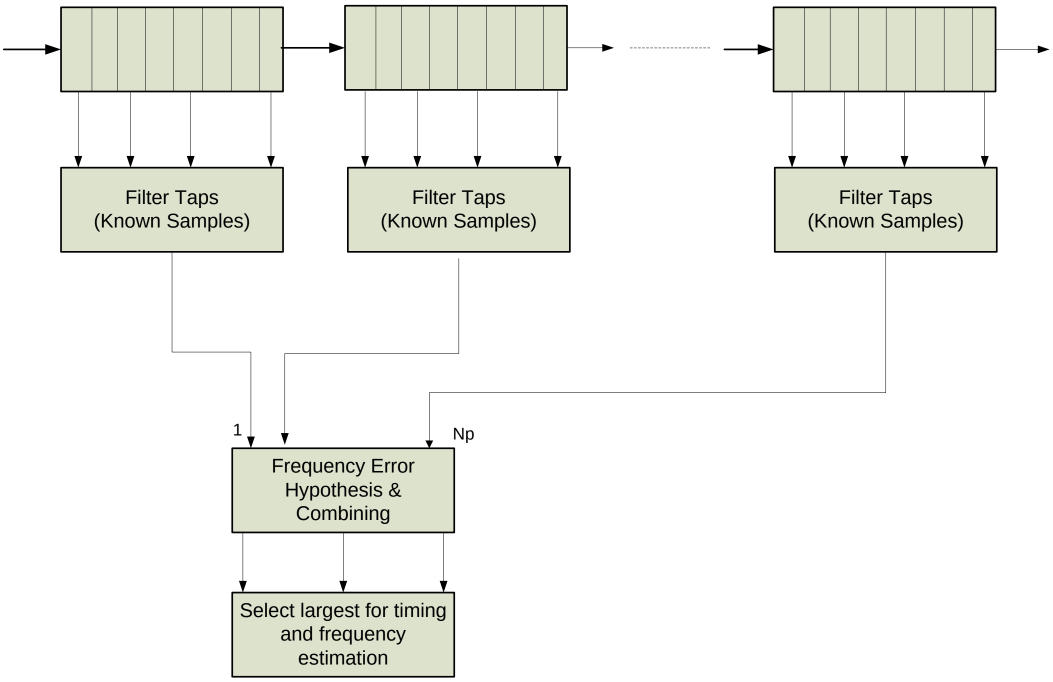

Filter Taps (Known Samples)
Filter Taps (Known Samples)
Filter Taps (Known Samples)
1
Np
Frequency Error Hypothesis & Combining
Select largest for timing and frequency estimation

Figure D.4: Burst Synchronizer for coarse timing and frequency estimations

D.4 Simulation results

Some simulation results with echo cancellation are reported below for two possible DVB-RCS2 LM waveforms. In particular, RCS2 Waveform ID=4 (QPSK ½) and waveform ID=8 (8PSK 2/3) were considered respectively for the QPSK and 8 PSK cases. Such waveforms are representative of short bursts in DVB-RCS2 and should be thus more critical as far as echo cancellation is concerned (with respect to longer bursts).

Figure D.5 shows results for QPSK1/2 (ID=4) with and without TSS both in linear and non-linear channel. For the non-linear channel a Rapp-Model with parameter s=6 was used for representing the terminal SSPA (satellite HPA is assumed to operate linearly). The Rapp model only considers the AM/AM non-linearity as it assumes that the SSPA AM/PM is negligible. The amplifier gain is modelled as:

The assumed signal baud rate was 128 kBaud and phase noise (if included) was compliant with guidelines of DVB-RCS. A frequency error of 2 % of the baud rate was also considered. As apparent from curves in Figure D.5, the loss caused by TSS in case of phase noise is limited to 0,2 dB only. The optimal SSPA IBO was 0 dB (at least for the selected Rapp model). For that IBO, the loss with respect to the linear case was 0,15 dB with TSS (against 0,1 dB in absence of TSS). Overall, the loss due to TSS in the worst case as far as phase noise, non-linearity and frequency error is only 0,2 dB (FER = 10^{-4}) with respect to the conventional case without TSS.

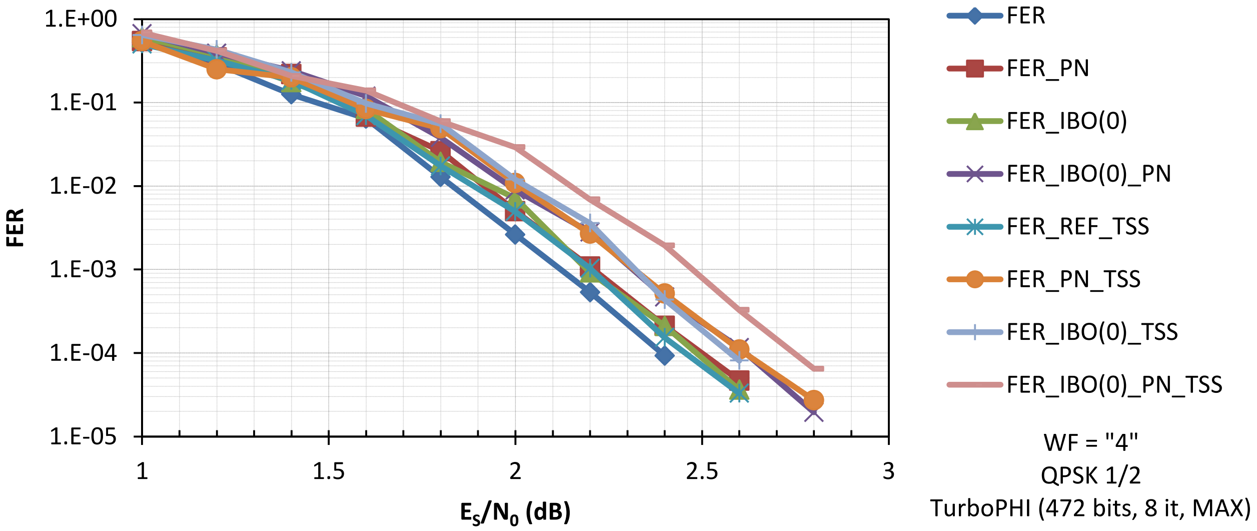


Figure D.5: Performance of DVB-RCS-2 QPSK ½ (Waveform ID=4) with and without TSS and different channel conditions

Figure D.6 shows the effect of carrier unbalance in TSS. This is justified by the fact that lower power echo, although cancelled less accurately, is clearly less disturbing. Conversely, a higher power echo is easier to cancel more accurately.

Figure D.7 shows results with 8PSK 2/3 (Waveform ID=8) under different channel conditions.

In presence of non-linearity (with IBO = 0 dB) TSS has about 0,65 dB penalization with respect to the conventional case. Even with perfect echo estimation, a residual interference would be present due to the non-compensated HPA distortion. This effect amounts to 0,2 dB (as can be inferred by comparison with and without TSS in a linear channel as shown in Figure D.8 and in non-linear channel as shown in Figure D.9). That 0,2 dB loss could be recovered if information about the non-linearity is available and exploited to reconstruct the echo.

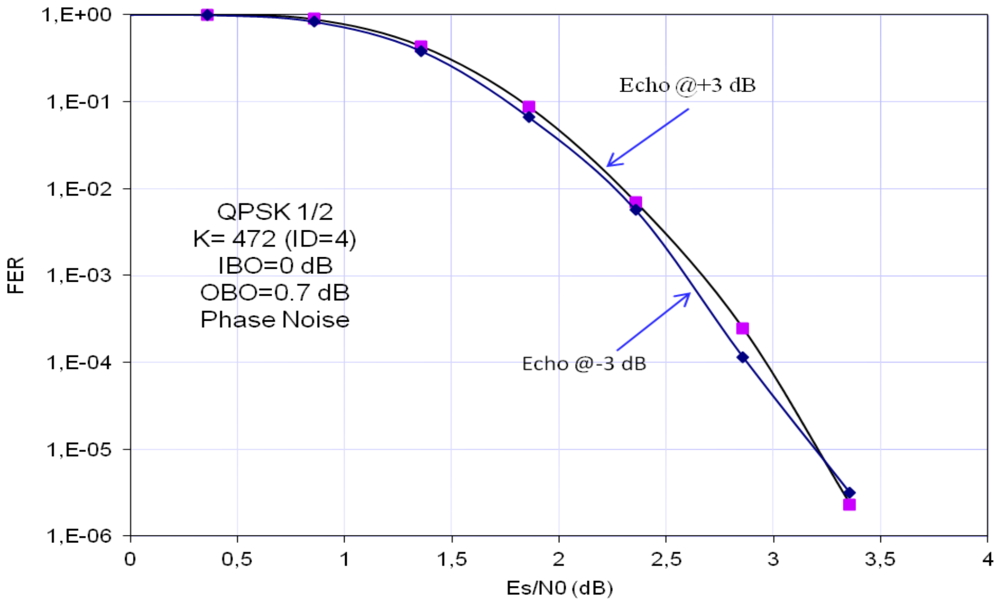


Figure D.6: Performance of Echo cancellation under carrier unbalance

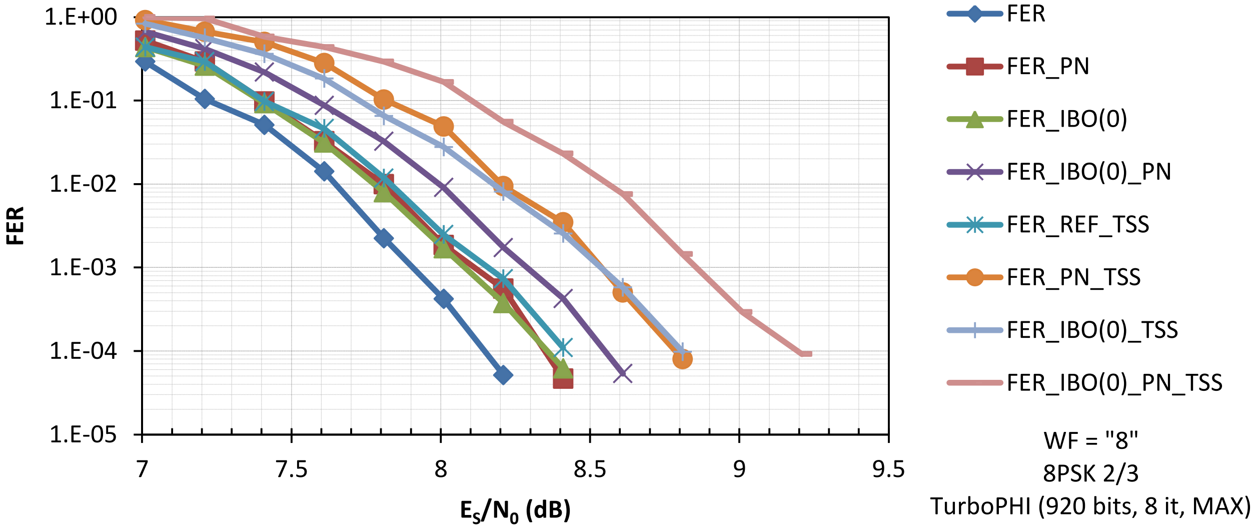
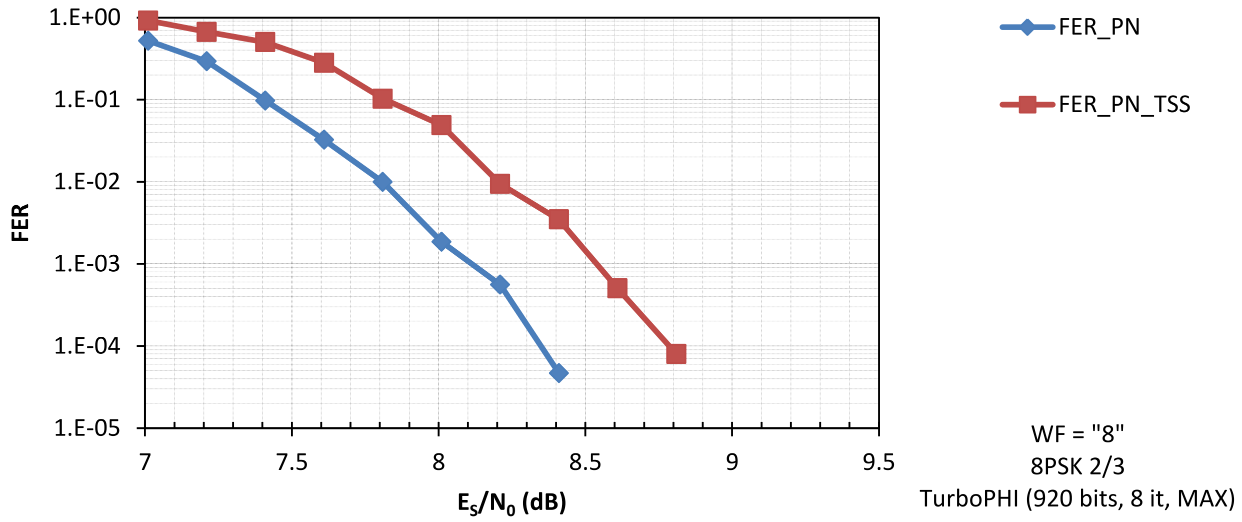
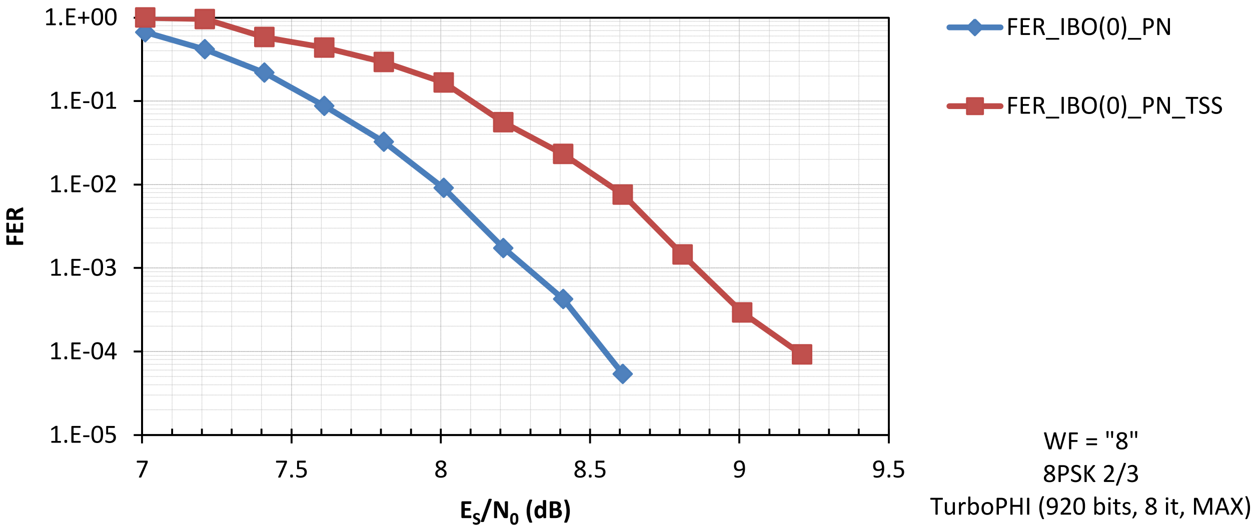


Figure D.7: Performance of waveform ID=8 (8PSK 2/3, short burst) under different channel conditions with and without TSS



NOTE: Losses due to TSS is about 0,4 dB.

Figure D.8: Comparison of performance in linear channel with and without TSS. Waveform ID=8 (8PSK 2/3, short burst), 128 kBaud, Phase Noise as in DVB-RCS guidelines



NOTE: Waveform ID=8 (8PSK 2/3, short burst), 128 kBaud, Phase Noise as in DVB-RCS guidelines. Losses due to TSS is about 0,65 dB.

Figure D.9: Comparison of performance in non-linear channel (IBO = 0 dB) with and without TSS

D.5 TSS Implementation

Support for Time Slot Sharing (TSS) can be included as an option within the signalling for dynamic mesh links control. Assuming that mesh signalling is supported by DCP, terminals can notify of TSS support with the DCP logon request message sent after logon. It should be noted that the name of "RCST capability request" (RCSTCapReq) is taken from current ETSI C2P specifications (TS 102 602 [i.75], "Connection Control Protocol for DVB-RCS"), although the message is clearly a notification. Anyway terminology in ETSI C2P is to name as "request" the originating messages and as "response" the related answer. Hence the corresponding message send by NCC in response to the RCST capability request" is "the RCST capability response". In this message, new IEs should be included for communicating various capabilities relevant to mesh networking (e.g. number of mesh receivers available in the RCST) as well as the support for TSS. The new IEs can be added in the range of User Defined IEs. The DCP messages allow the introduction of additional IEs.

After the NCC has been informed about "Time Slot Sharing" support, the usage of it may be managed either statically or dynamically:

- for static management, it is sufficient the NCC knows if an RCST is "Time Slot Sharing capable": for every connection requested, if the peer RCST is also "Time Slot Sharing capable", the connection is assigned by NCC as "Time Slot Sharing type";
- for dynamic management, an RCST "Time Slot Sharing capable", after signalling to NCC, is required to request for each connection if it wants to use "Time Slot Sharing";

In the last case, at mesh link service establishment, an RCST should explicitly signals if it wants to use "Time Slot Sharing" for the new connection request, adding the new "Time Slot Sharing" IE to the Link Service Establishment Request message; the peer RCST should be informed accordingly by the NCC.

Also for this case, the NCC may assign burst by burst resources in "Time Slot Sharing" mode in an implicit way through the TBTP2 (as the same slot may be assigned to more than one terminal). This clearly requires that each terminal detect this condition by comparing the list of slots to receive (i.e. those assigned to its peer for transmission) with those in which he has transmitted.

The "Time Slot Sharing" IE should also include additional fields to allow one of different policies (and their parameter, if any) to be specified.

An example of "Time Slot Sharing" connection establishment for dynamic management case is described below:

- after the logon, an RCST sends DCP logon request message with "Time Slot Sharing" IE (for specific details on "Time Slot Sharing" usage);
- an initiating RCST sends Link Service Establishment Request to NCC with "Time Slot Sharing" IE (if it wants to use now "Time Slot Sharing");
- the NCC verify if the peer is "Time Slot Sharing" capable: if so, sends Link Service Establishment Request to the peer with "Time Slot Sharing" IE;
- after that, the NCC sends Link Service Establishment Response to the originating RCST with "Time Slot Sharing" IE;
- the broadcasted TBTP2s should assign the same slot to both parties. Slots to be used in TSS should be associated to a specific resource identifier (which may be different for each the terminal and is assigned by the NCC at the mesh link service establishment request as an `assignment_id` similar to those assigned at logon by the NCC at every terminal) which has been exclusively allocated for that peer-to-peer connection.

The "Time Slot Sharing" technique may be extended to allow a partial channel sharing also in multicast multipart video-conferences with 3 or 4 participants max (only for multicast contributions) if all participants are "Multicast Time Slot Sharing capable".

The bandwidth gain will reduce with increasing participant number, so for more than 4 participants (25 % bandwidth gain) it is no more practical continue to use this technique.

A multiparty video-conference is typically established by other higher level (usually at application level) signalling which is also in charge of managing the communication of all relevant parameters (video/audio format, speed, etc.) and the multicast addresses of all participants.

In case of "Multicast Time Slot Sharing" with more than 2 participants, the wanted signal is recovered with iterative multiple cancellations at physical layer.

Finally TSS also requires that signals transmitted in the same time slot are uncorrelated for correct cancellation. Different preambles and scrambling sequences should be used by each of the two peer terminals to uncorrelate the peer transmitted bursts. At present, this is not supported by the RCS2 lower layer specs as preamble are defined per time slot; also scrambling is fixed for all users and is not applied on the preamble. In order to not change current specs, we may assume that the "Time Slot Sharing" IE sent by the NCC to terminals also contain a flag which may change the semantic of some physical layer spec. In particular such flag, if active, could change the order with which preamble symbols are transmitted (i.e. one of the peer would transmit a time mirrored preamble instead of the normal one) and the initialization word of the burst scrambler.

Annex E: Forward Link Spectrum Spreading

E.1 Introduction

The normative document (see clause 5.4.5 of [i.1]) introduces the optional use of spread spectrum techniques for the forward link. The use of spread spectrum is particularly useful for applications such as interactive services for mobile RCSTs with small antenna aperture as well as extending the operating ranges in the presence of severe atmospheric fading conditions.

Two alternative solutions for the forward link spectrum spreading are introduced, namely Direct Sequence Spectrum Spreading (DSSS) and Frame Repetition Spectrum Spreading (FRSS). These techniques have their strengths and weaknesses that make them more suitable for different applications as summarized in Table E.1.

Table E.1: Two Forward Link Spread Spectrum Techniques

Spectrum Spreading Technique	Strong Points	Weak points
Direct Sequence Spectrum Spreading	- Support of very lower SNR - Robust against carrier instability	- Not compatible with DVB-S2 waveform. Therefore, cannot share the same carrier with a conventional DVB-S2 service.
Frame Repetition Spectrum Spreading	- Coexistence with non-spread DVB-S2 waveform [i.2] (backward compatibility)	- Impact of carrier instability

E.2 Technical Description

E.2.1 Direct Sequence Spectrum Spreading

A DVB-S2 forward link transmission based on symbol repetition can be spread in bandwidth using the provisions in this clause. Such spreading is applied in two stages: spreading and scrambling. The first operation, spreading, multiplies every (I+jQ) PL symbol by a sequence of chips to enlarge the bandwidth of the signal (the PL frame duration in absolute time remains the same). The number of chips per symbol is called the Spreading Factor (SF). The waveform bandwidth expansion is proportional for SF. When SF=1, the transmission is equivalent to a conventional DVB-S2.

The second operation, scrambling, applies a scrambling code to the spread signal. The processing is illustrated in Figure E.1.

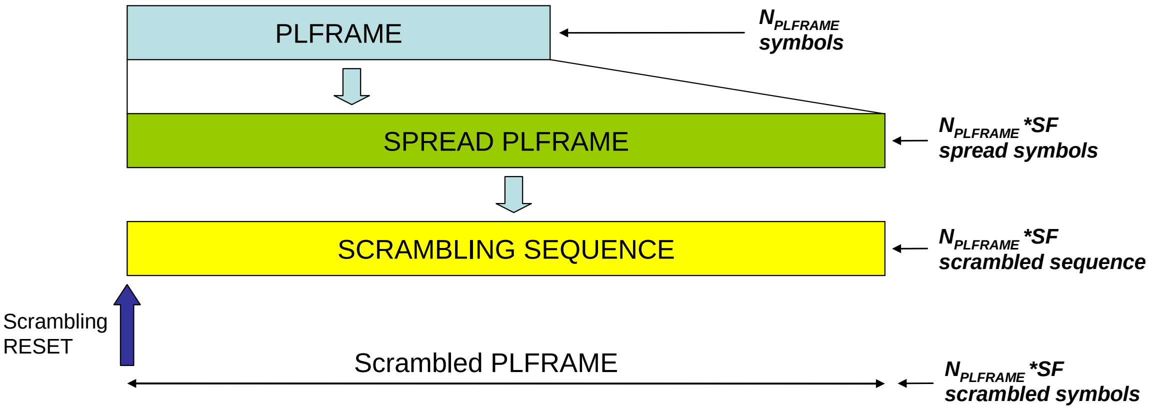

PLFRAME
$N_{PLFRAME}$ symbols
SPREAD PLFRAME
$N_{PLFRAME}$ *SF spread symbols
SCRAMBLING SEQUENCE
$N_{PLFRAME}$ *SF scrambled sequence
Scrambling RESET
Scrambled PLFRAME
$N_{PLFRAME}$ *SF scrambled symbols

Figure E.1: Direct Sequence Spreading Technique

The spectrum spreading should be applied on a PLFRAME basis, following the conventional DVB-S2 physical layer scrambling process. Each symbol in a PLFRAME, including the PLHEADER and pilot symbols, if used, should be spread by a repetition of a real-valued spreading code *C* (*i*). The output of the spreading for each symbol on the *I* and *Q* branches should thus be a sequence of *SF* chips corresponding to the spreading code chip sequence, multiplied by the corresponding, real-valued symbol component value. The spreading code sequence should be time-aligned with the symbol boundary.

If $\{d[k]\}$, $k = 0, 1,\ldots, N_{PLFRAME}-1$, represents the $(I+jQ)$ symbols of the PLFRAME, where $N_{PLFRAME}$ is the number of symbols in one PLFRAME, then the spreading operation yields the spread sequence $s(i)$:

$$s(i) = d(\lfloor i/SF \rfloor)C(\mathrm{mod}(i,SF)) \qquad \text{for } i = 0, 1,..., (N_{PLFRAME} \times SF)-1$$

Spreading codes $C(i)$ are defined for spreading factors of 1, 2, 3 and 4 and are signalled in the extended Satellite Forward Link Descriptor in clause 6.4.17.6 (Table 6-50 of [i.1]). The extended signalling description is given in Table E-2.

In terms of the reference modulator signal flow defined in DVB-S2 [i.2], the spreading should be performed immediately prior to the physical layer scrambling.

The second operation, scrambling, is achieved through the use of the same method as that defined for physical layer scrambling in clause 5.5.4 of [i.2], except that:

- the length of the scrambling sequence is here equal to $N_{PLFRAME} \times SF$, rather than $N_{PLFRAME}$; and
- the scrambling sequence is applied to the entire spread PLFRAME, including the PLHEADER and pilots if used.

The scrambling sequence should be aligned with the PLFRAME epoch, and it should be re-initialized at the beginning of each PLFRAME.

The sequence of complex valued chips is scrambled (complex chip-wise multiplication) by the complex-valued scrambling code, $w(i)$, defined in clause 5.5.4 of [i.2], when *SF* is greater than 1. The Spread PLFRAME duration depends on the selected modulation and the adopted spreading factor. The scrambled symbols, $z(i)$, is obtained by directly multiplying the spread symbols, $s(i)$, by the scrambling sequence, $w(i)$, as follows:

$$z(i) = s(i) \times w(i \text{ modulo } 66420),\ i=0,1,2,\ldots,(N_{PLFRAME} \times SF)-1$$

After scrambling, the signal $\{z(i)\}$ is filtered using square root raised cosine filter with a pre-selected roll-off factor as described in clause 5.6 of [i.2].

It is necessary to define an explicit scrambling sequence in the corresponding satellite forward link descriptor (clause 6.4.17.6 of [i.1]) when *SF* is greater than 1.

Table E.2: Syntax of the Satellite Forward Link descriptor for Direct Sequence Spread

Syntax	No. of bits		Information Mnemonic
	Reserved	Information	
Satellite_forward_link_descriptor() {			
descriptor_tag		8	uimsbf
descriptor_length		8	uimsbf
satellite_ID		8	uimsbf
beam_ID		16	uimsbf
NCC_ID		8	uimsbf
multiplex_usage		3	bslbf
local_multiplex_ID		5	uimsbf
frequency		32	uimsbf
orbital_position		16	bslbf
west_east_flag		1	bslbf
Polarization		2	bslbf
transmission_standard		2	uimsbf
if (transmission_standard == 0) {			
"001"		3	bslbf
}			
else if ((transmission_standard == 1) or (transmission_standard == 2)) {			

Syntax	No. of bits		Information Mnemonic
	Reserved	Information	
scrambling sequence selector		1	bslbf
roll_off		2	uimsbf
}			
symbol_rate		24	uimsbf
if (transmission_standard == 0){			
FEC_inner		4	bslbf
Reserved	4		bslbf
}			
else if ((transmission_standard == 1) or (transmission_standard == 2)) {			
Input_Stream_Identifier		8	uimsbf
if (scrambling_sequence_selector == 0)			
spreading_code_selector		3	
scrambling_sequence_index	3	18	uimsbf
}			
for (i=0; i<N; i++) {			
private_data_byte		8	bslbf
}			
}			

Semantics for the Satellite_forward_link_descriptor:

- spreading_code_selector: This 3-bit field defines the chip sequence used to achieve spectrum spreading, in accordance with Table E.3.

Table E.3: Forward link spreading sequences for Direct Sequence

Value	Spreading factor	Chip sequence
000	2	1, 1
001	2	1, -1
010	3	1, 1, 1
011	4	1, 1, 1, 1
100	4	1, 1, -1, -1
101	4	1, -1, 1, -1
110	4	1, -1, -1, 1
111	1 (no spreading)	1

E.2.2 Frame Repetition Spectrum Spreading

A DVB-S2 forward link transmission can be spread in bandwidth based on PL-frame repetition using the provisions described in this clause.

The frame repetition spectrum spreading is applied in four stages (see example in Figure E.2 for *SF*=2):

1) $\pi/2$ BPSK symbol mapping from short LDPC codeword as the same symbol mapping method of PLHEADER block defined in clause 5.5.2 of DVB-S2 [i.2], PLFRAME is constructed with addition of PLHEADER block.

2) Xspread Frame is constructed through PLFRAME repetition according to the number of SF(Spreading Factor) that is called as the number of identical PLFRAME to enlarge the bandwidth of the signal. When SF=1, the transmission is a conventional DVB-S2 signal that can transmit $\pi/2$ BPSK modulation.

3) Scrambler applies a scrambling sequence to the Xspread Frame. Spreading is applied on $\pi/2$ BPSK Xspread Frame basis, following the DVB-S2 physical layer scrambling process defined in clause 5.5.4 of [i.2]. The scrambling process is applied to the Xspread Frame.

4) After scrambling process, Spread Frame is constructed through recombination of a pair of PLHEADER and Xspread Frame fragmented from Xspread Frame in Figure E.2.

The value of SF, i.e. 1, 2 and 3, is signalled in the extended Satellite Forward Link Descriptor in clause 6.4.17.6 of [i.1] and in the modified MODCOD table as shown in Table E.4. The extended signalling description is given in Table E.5. The Spread Frame is transmitted using a π/2 BPSK modulation. Code rates 1/4 and 1/3 with a short LDPC codeword and no pilot mode are considered. When spread signal is transmitted, the MODCOD table (Table 12 of DVB-S2 [i.2]) should be replaced with Table E.4 as shown below.

Specifically, in case of the specified MODCOD of 16APSK corresponding to 18_D,19_D,20_D,21_D,22_D and 23_D, non-pilot and short frame size mode, the relevant MODCOD and TYPE field is used for spread signal transmission in coexistence network with spread and non-spread signal. After Spread Frame construction, the signal is filtered using a squared root raised cosine filtered with a pre-selected roll-off factor as described in clause 5.6 of [i.2].

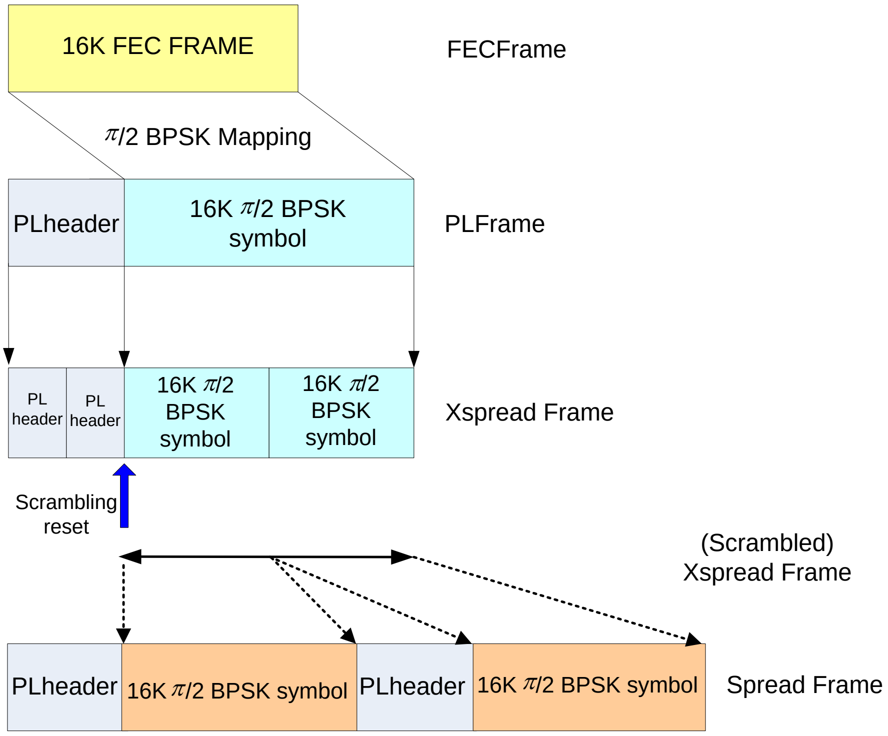


Figure E.2: Forward link spectrum spreading (in case of SF 2)

Table E.4: MODCOD coding for spreading signal

Mode	MODCOD	The MSB of the TYPE field(16K/64K)	The MSB of the TYPE field(Pilot/Nonpilot)
π/2BPSK 1/4 and Spreading Factor(SF) 1	18_D	1	0
π/2BPSK 1/3 and Spreading Factor(SF) 1	19_D	1	0
π/2BPSK 1/4 and Spreading Factor(SF) 2	20_D	1	0
π/2BPSK 1/3 and Spreading Factor(SF) 2	21_D	1	0
π/2BPSK 1/4 and Spreading Factor(SF) 3	22_D	1	0
π/2BPSK 1/3 and Spreading Factor(SF) 3	23_D	1	0

Table E.5: Syntax of the Satellite Forward Link descriptor for Frame Repetition

Syntax	No. of bits		Information Mnemonic
	Reserved	Information	
Satellite_forward_link_descriptor() {			
descriptor_tag		8	uimsbf
descriptor_length		8	uimsbf
satellite_ID		8	uimsbf

Syntax	No. of bits		Information Mnemonic
	Reserved	Information	
beam_ID		16	uimsbf
NCC_ID		8	uimsbf
multiplex_usage		3	bslbf
local_multiplex_ID		5	uimsbf
frequency		32	uimsbf
orbital_position		16	bslbf
west_east_flag		1	bslbf
Polarization		2	bslbf
transmission_standard		2	uimsbf
if (transmission_standard == 0) {			
"001"		3	bslbf
}			
else if ((transmission_standard == 1) or (transmission_standard == 2)) {			
scrambling sequence selector		1	bslbf
roll_off		2	uimsbf
}			
symbol_rate		24	uimsbf
if (transmission_standard == 0){			
FEC_inner		4	bslbf
Reserved	4		bslbf
}			
else if ((transmission_standard == 1) or (transmission_standard == 2)) {			
Input_Stream_Identifier		8	uimsbf
if (scrambling_sequence_selector == 0)			
spreading_code_selector		3	
scrambling_sequence_index	3	18	uimsbf
}			
for (i=0; i<N; i++) {			
private_data_byte		8	bslbf
}			
}			

Table E.6: Forward link spreading sequences for Frame repetition

Value	Spreading factor	MODCOD
000	1	π/2BPSK + code rate 1/4
001	1	π/2BPSK + code rate 1/3
010	2	π/2BPSK + code rate 1/4
011	2	π/2BPSK + code rate 1/3
100	3	π/2BPSK + code rate 1/4
101	3	π/2BPSK + code rate 1/3
110	reserved	1, reserved
111	1 (no spreading)	Conventional DVB-S2 waveform (except π/2BPSK)

E.3 Performance Evaluation

E.3.1 Direct Sequence Spectrum Spreading

The impact of Direct Sequence Spread Spectrum on the Frame Error Ratio has been investigated. In particular, the following system parameters have been considered:

- MODCOD: 1/4-QPSK
- Chip rate = 27,5 Mchip/sec

- Symbol rate = Chip rate / Spreading factor
- Train speed = 300 km/h
- Propagation channel: AWGN and correlated Ricean channel with Rice factor = 17 dB
- Satellite HPA IBO = 0,5 dB
- No interference from adjacent satellites

The robustness of the DVB-S2 spread signal with respect to non-linear distortion by different IBO values of satellite HPA is presented in Figure E.3 in an AWGN channel without mobility effect. The comparison between the spread and non-spread signals indicates that the signal spreading slightly improves the physical layer performance in the presence of a non-linear distortion.

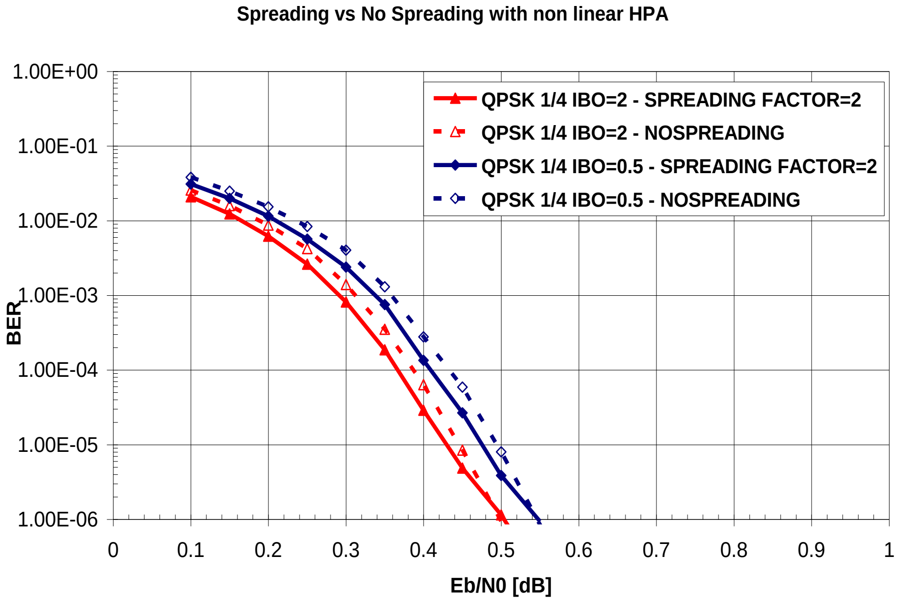


Figure E.3: Comparison between spread and not spread signal in AWGN channel in the presence of non-linear distortion

The impact of terminal mobility and nonlinear HPA effect on the performance has been analysed. Results are presented in Figure E.4. The same chip-rate (27,5 Mchip/sec) has been considered, thus the transmission symbol rate is 13,75 MBaud and 6,85 MBaud for SF=2 and SF=4, respectively. As noted before, spectrum spreading introduces a slight gain with respect to that of a conventional signal. The real benefit of the spreading factor is in the link margin, showing a gain of 3 dB and 6 dB for spreading factor 2 and 4, respectively.

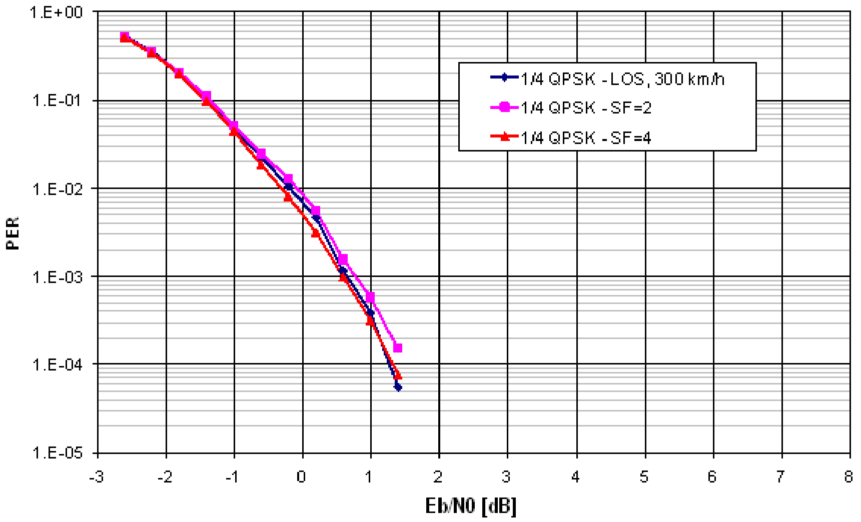


Figure E.4: Comparison between spread and not spread signal in LOS channel with the presence of non-linear HPA

E.3.2 Frame Repetition Spectrum Spreading

The impact of Direct Sequence Spread Spectrum on the Frame Error Ratio has been investigated. In particular, the following system parameters have been considered:

- MODCOD: 1/4-π/2BPSK
- Symbol rate = 1 Msymbol/sec and 10 Msymbol/sec
- Train speed = 300 km/h
- Propagation channel: AWGN and correlated Ricean channel with Rice factor = 17 dB
- Satellite HPA IBO = -1 dB
- Phase noise (DVB-S2 typical)
- No interference from adjacent satellites

The performance of the DVB-S2 spread signal based on frame repetition scheme is presented in Figure E.5 with all channel impairment except adjacent channel interference. The comparison between the spread and non-spread signals highlights that the signal spreading slightly deteriorates as spread factor increases and symbol rate and SNIR decreases.

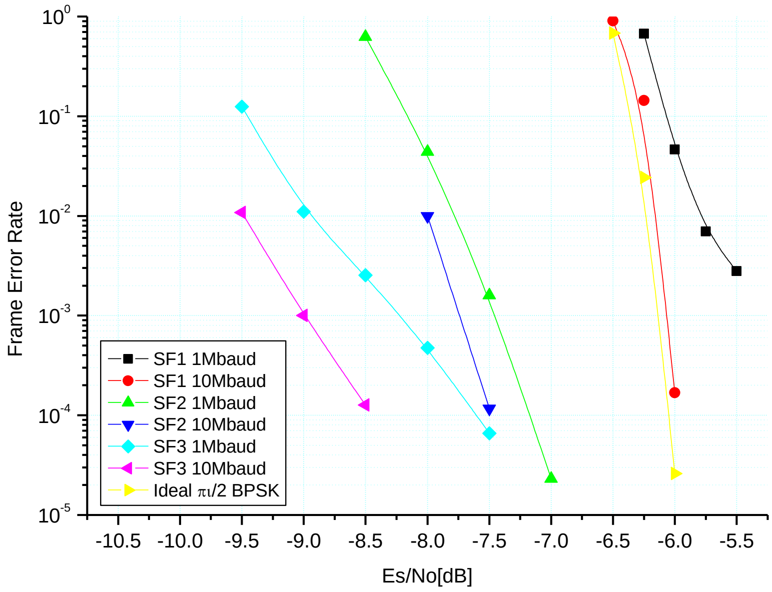


Figure E.5: Frame Error Ratio performance for Frame Repetition Spectrum Spreading

Annex F: Return Link Spectrum Spreading By Burst Repetition

In Mobile LOS scenarios, return link carriers may require the use of spectrum spreading in order to reduce the spectral density and in particular off-axis EIRP emission density.

The normative document [i.1] specifies as an optional feature the use of direct-sequence spread spectrum link to achieve spectrum spreading in the return link.

The first generation of DVB-RCS supports an alternative solution for spectrum spreading based on burst repetition. This technique has a minimum impact on the RCST hardware implementation. Burst repetition can be employed for certain special applications, but is not recommended for general spread-spectrum transmission purposes due to possible performance degradation (compared to direct-sequence spread spectrum techniques). This clause describes the burst repetition technique and implementation aspects as a user defined feature.

F.1 Spreading description

The burst repetition consists of increasing the symbol rate of the signal by a factor N without increasing the power. This modification reduces the E_s/N_0 at the receiver side. In order to recover the required E_s/N_0, each burst is repeated *N* times at the transmitter side. The signal after spreading is depicted in Figure F.1.

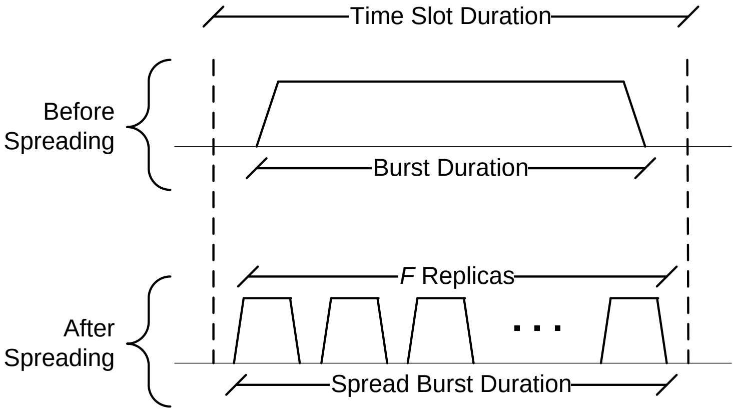


Figure F.1: Spectrum spreading by burst repetition

In order to avoid discrete lines in the output spectrum, a random phase shift should be applied to each replica before transmission. This has no impact on the de-spreading since the relative phases between replicas are estimated in the receiver.

F.2 De-spreading description

De-spreading is achieved by summing together corresponding signal samples of successive replicas.

Prerequisite of the approach is that the timing error and Doppler of the transmitter with respect to the gateway is not such that a significant drift of the sampling time at the gateway happens during the reception of the replicas to be combined. This is actually the case in typical operation scenarios. In fact, even considering the worst case Doppler experienced in the aeronautical domain (1 100 ns/s), more than of 45 000 symbols would be required for timing errors to accumulate up to value larger than 10 % of the symbol period.

Obviously combining should be such that homologous samples are added in phase; therefore a phase alignment is required before combining. The principle for the recombination is shown in Figure F.2.

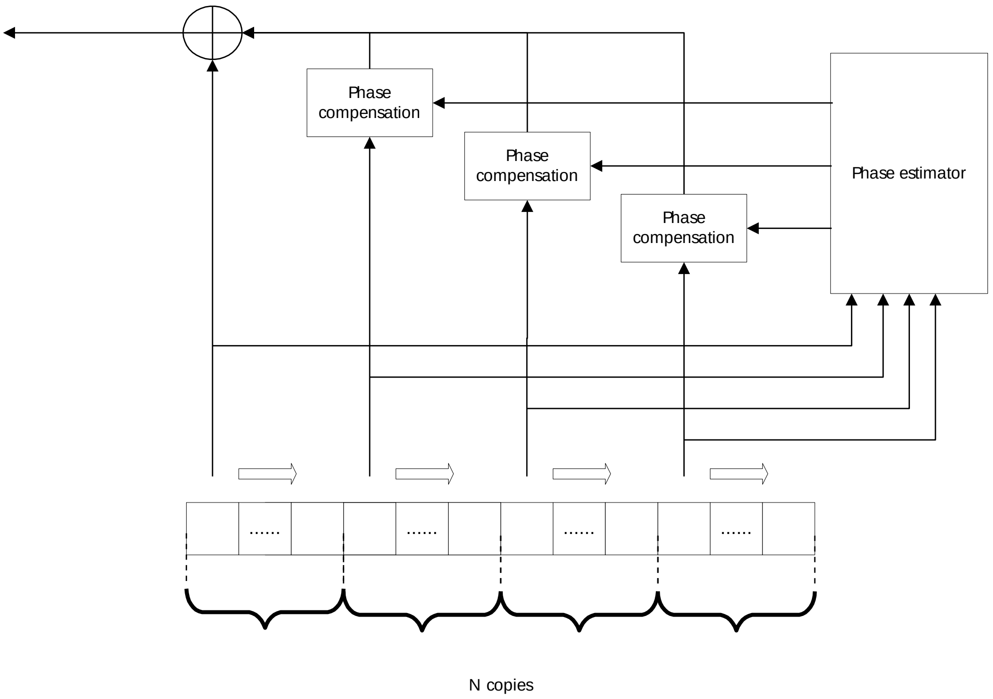


Figure F.2: Coherent recombination

The de-spreading solution by recombination needs to memorize at least 2 replicas in order to perform the relative phase estimation. This estimation can be done via a block correlator because what we need is exp(jΔϕ) and not Δϕ itself. An iterative recombining scheme will minimize the needed memory.

In order to have a good phase alignment, the differential frequency shift between replicas should be as low as possible. The corresponding frequency shift tolerance will depend on the burst size.

The frequency tolerance and the phase shift tolerance are related by the following formula:

$$\Delta f \cdot T_s = \frac{\Delta\varphi}{\pi \cdot N_S}$$

where N_s is the number of symbols inside a replica, T_s is the symbol duration after spreading, Δφ is the maximal phase tolerance and Δf is the frequency tolerance between replicas. The longer is the burst, the lower is the frequency tolerance.

Typically, in order to have negligible degradations, a 15° error phase between replicas have been considered. Considering a 512 kBaud carrier and a 15° tolerance, the maximal frequency tolerance between 2 consecutive replicas is 45 Hz for a burst of 1 000 symbols length. This value is compatible with a 1 kHz/s frequency drift in Ku-Band.

After the de-spreading, the estimations of the absolute delay, frequency and phase are done by conventional timing and carrier recovery algorithms, so a classical DVB-RCS demodulator can be used as shown in Figure F.3.

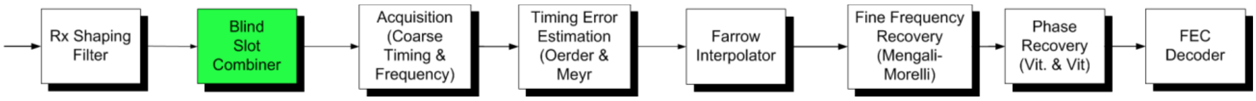

Rx Shaping Filter
Blind Slot Combiner
Acquisition (Coarse Timing & Frequency)
Timing Error Estimation (Oerder & Meyr
Farrow Interpolator
Fine Frequency Recovery (Mengali-Morelli)
Phase Recovery (Vit. & Vit)
FEC Decoder

Figure F.3: Receiver architecture including dispreading

When spectrum spreading by burst repetition is applied, the transmission instants of individual replicas of the same burst should not vary relative to the start instants of their respective sub-timeslots by more than ±5 % of the symbol period.

F.3 Burst Repetition Implementation

The necessary signalling is supported in the BCT in a user defined field.

F.3.1 Description of FL L2S Components

Table 6-15 of the normative document [i.1] defines all transmission types used in the frame type. In order to support the burst repetition, a new entry needs to added to this table as shown in Table F.1:

- `tx_format_class`: This field indicates the transmission format class of all transmission types used in the frame type. The values are assigned in Table F.1.

Table F.1: Coding of Transmission Format Classes

Value	tx_format_class
0	Reserved
1	Linear Modulation Burst Transmission
2	Continuous Phase Modulation Burst Transmission
3	Continuous Transmission
4	Spread-Spectrum Linear Modulation Burst Transmission
5 to 127	Reserved
128	User Defined (Generic Continuous Phase Modulation Burst Transmission)
129	Burst Repetition
130 to 255	User Defined

Table 6-26 of the normative document [i.1] defines the data block for the return link spreading. Similar description for the burst repetition data block can be defined:

- modulation_scheme: This is an 8-bit field which serves as an identifier of the modulation scheme as defined in Table F.2. When spread-spectrum transmission is employed, the three LSB of the field indicate the spreading factor as defined in Table F.3.

Table F.2: Modulation Scheme Code Values

Modulation Scheme	Value
Reserved	0x00 to 0x04
π/2-BPSK (No Spreading)	0x05
Reserved for future use	0x06 to 0x0f
π/2-BPSK with burst repetition spreading	0x10 to 0x17
QPSK with burst repetition spreading	0x18 to 0x1f
π/2-BPSK with direct-sequence spreading	0x20 to 0x27
Reserved	0x28 to 0x7f
User defined	0x80 to 0xff

Table F.3: Return link spreading factors

Spreading factor	Modulation Scheme LSB's
2	000
3	001
4	010
6	011
8	100
10	101
13	110
16	111

Annex G: RCST IDU/ODU Cable IFL Protocol Description

This clause provides a description of a cable protocol between the IDU and ODU where the IDU acts as a Master and the ODU acts as a Slave. The protocol is based on an extension of the Eutelsat DiSEqC™ bus specification Version 4.2 [i.20]. This protocol was also reported in annex B of the implementation guidelines for the first generation of the DVB-RCS systems.

The control and management of a DiSEqC™ ODU is not completely defined by this specification and system dependent issues should be expected. There are as well optional protocol elements. It should be expected that an IDU need some special adaptation to partly and fully exploit a specific type of DiSEqC™ ODU.

G.1 Command and request processing

Only one command or status request can be processed at a time. Once the IDU has issued a command or status request to the ODU, a new command cannot be issued until the IDU has received a valid response (ACK or NACK) or the command has timed out. In the case of either a NACK or time-out, the IDU may issue a given command up to three times before declaring a fault on the interface.

G.2 Alarms

When a hardware alarm occurs within the ODU, the ODU should:

1) disable the SSPA to inhibit transmission by removing power to the Tx circuit;

2) disable the frequency reference signal provided to the IDU (if implemented);

3) buffer the fault indication until read or cleared by the IDU.

After detecting the ODU fault (via loss of reference, hub report of abnormal logoff or time-out of command request), the IDU should send a status request message to the ODU to identify the type of alarm.

G.3 Dynamic behaviour

Unless otherwise specified in the subsequent clauses, the ODU should respond to a command or request received from the IDU within the allowable timeout period (T_{OODU}) of 150 ms. In general, to force the ODU to respond immediately to a command/request from the IDU, the DiSEqC™ command 0x01 may be sent after any power-on or (re-) initialization procedure.

NOTE: DiSEqC™ devices will by default apply a random response time between 15 ms and 115 ms, and even 135 ms in case of need for collision avoidance.

The T_{OODU} may be disabled during installation.

G.4 Error recovery mechanism

When the IDU does not receive its expected answer (no answer or NACK), it may re-send the message twice, after which an alarm should be raised. From the ODUs point of view, there will be no limitations on the number of times that the IDU can attempt to send a message. If the ODU keeps receiving messages in error, it will continually respond with the error code. If there is an invalid password, and the ODU requires command authorization, it will "lock-up" after the sixth attempt (see clause G.5.4).

G.5 Message level description

The Monitoring and Control Protocol message is depicted in Figure G.1. A dedicated protocol will be used for extended messages longer than 8 bytes (e.g. software downloads). It is described in detail in clause G.5.5. Bytes are transmitted MSB first, and each byte is followed by an odd parity bit.

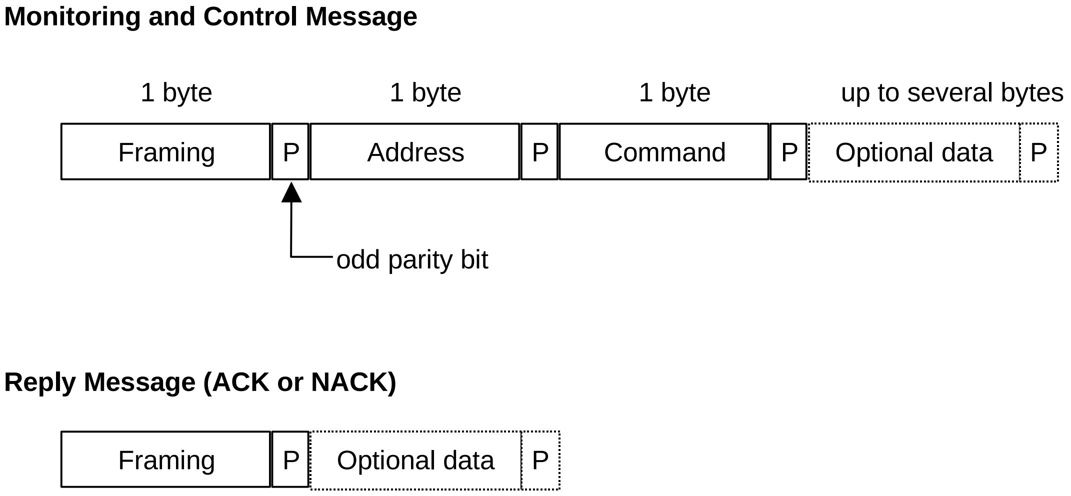


Figure G.1: Message format

"Framing", "Address" and "Command" fields are detailed in clauses G.5.1, G.5.2 and G.5.3 accordingly.

G.5.1 Framing field description

The Framing Field is described in Table G.1.

Table G.1: Framing definitions

Hex Byte	Binary	Framing byte Function
0xE2	1110 0010	Command from Master, Reply required, First transmission
0xE4	1110 0100	Reply from Slave, "OK", no errors detected
0xE5	1110 0101	Reply from Slave, Command not supported by slave
0xE6	1110 0110	Reply from Slave, Parity Error detected - Request repeat
0xE7	1110 0111	Reply from Slave, message format not recognized - Request repeat.
0xE8	1110 1000	Extended Command from Master, Reply required only after last message block, First transmission
0xE9	1110 1001	Extended Command from Master, Reply required only after last message block, Repeated transmission
0xEA	1110 1010	Extended Command from Master, Reply required after each message block, First transmission
0xEB	1110 1011	Extended Command from Master, Reply required after each message block, Repeated transmission
0xEC 00	1110 1100 0000 0000	Reply from Slave, command understood, task not yet completed, unknown time to execute
0xEC nn	1110 1100 nn	Reply from Slave, command understood, check if task completed after nn seconds (1 to 127 binary)
0xED nn	1110 1101 1111 nnnn	Reply from Slave, repeat block nn (where nn is between 01 and 2C)
0xED E1	1110 1101 1110 0001	Reply from Slave, EUI-64 of IDU not valid
0xED Fp	1110 1101 1111 pppp	Reply from Slave, relating to password commands, where p indicates:
0xED F0	1110 1101 1111 0000	Reply from Slave, password in the incoming string not valid (attempt 1 to n - unidentified number of non-critical attempts)
0xED Fn	1110 1101 1111 nnnn	Reply from Slave, password in the incoming string not valid (n identifies the sequencenumber of a non-critical failing attempt)
0xED FE	1110 1101 1111 1110	Reply from slave, password in the incoming string not valid, pen-ultimate attempt (e.g. attempt 5)
0xED FF	1110 1101 1111 1111	Reply from slave, ODU Locked (installer required - 6 or more attempts made in a row applying wrong password)
0xEE 00	1110 1110	Reply from Slave, CRC not valid (no additional information)
0xEF	1110 1111	Reply from Slave, additional blocks to follow
0xF0	1111 0000	Request from Slave
0xF1	1111 0001	Reply from Master, OK
0xF2	1111 0010	Reply from Master, Error
NOTE: The framing commands are grouped in pairs, where the value of the 2nd LSB of the first bytes gives an indication whether a further response is expected ("1") or not ("0"), although this is not a "hard" rule this should assist with low level detection software.		

A positive acknowledgement (0xE4) should be used to reply that the message from the Master has been successfully received. Any data requested by the IDU (defined by the original command from the IDU) will be sent directly after the positive acknowledgement byte. If the reply is more than 8 bytes in total then it should use the extended message structure (see clause G.5.5.3).

Negative acknowledgements should use values 0xE5, 0xE6 and 0xE7 (no additional data is permitted) as defined in Table G.1.

0xE5 should additionally be used when a command is not supported or cannot be implemented due to a functional problem in the ODU. In this case the ODU should also flag an alarm to the IDU using the mechanism described in clause G.2.

0xE6 Parity error detected (this will, in practice, occur as the result of transmission error), parity check is performed on each byte of each command. Notice that in case of CRC error, the reply is 0xEE.

0xE7 should be used to flag an incompatibility between a command and any other field rendering execution of that command impossible for example incorrect message structure, wrong number of bits or bytes.

In cases where the password is used in the command, the ODU may reply with 0xEB, 0xEC, 0xED or 0xEE.

G.5.2 Address field description

This field (encoded in one byte) specifies the destination subsystems for each message according to definitions in Table G.2.

Table G.2: Address definitions

Hex Byte	Binary	Address byte Function
0x80	1000 0000	Any RCST (and VSAT)
0x81	1000 0001	IDU of RCST
0x82	1000 0010	ODU of RCST
0x83	1000 0011	Other (Future extension)

G.5.3 Command field description (IDU → ODU)

This field (encoded in one byte) specifies the required action for the addressed subsystem according to definitions in Table G.3.

Table G.3: Command definitions

Hex. Byte	Use password	Command	Status	Action
00	No	Reset	O	Reset all the ODU functions (same as power down reset)
0A	No	Soft Reset	O	ODU Software Reset
12	No	Monitoring	M	Request the general status of the ODU
5C	No	Manufacturer's ID	M	Request Manufacturer's Identification
5D	No	Product ID	M	Request the Product's Identification
C1	No	Download start	O	Allow the ODU to enter into download mode
C2	No	Download data	O	Download software data
C3	No	Download abort	O	Abort the download process
C4	No	Download valid	O	Grab the "new" software into a non volatile memory
C5	No	Download toggle	O	Toggle between current software version and previous one
C6	Optional	SSPA ON	M	Enable the ODU amplification output
C7	No	SSPA OFF	M	Disable the ODU amplification output
C8	Optional	Set power level	O	Set SSPA power level
C9	No	Mod ON	O	Normal operation
CA	Optional	Mod OFF	O	Modulation Off, transmit Continuous Wave (CW)
CB	Yes	Change password	O	Enable the ODU password modification
CC	Yes	Validate password	O	Activate the new password
CD	Yes	Reset ODU locked	O	Reset the "Faulty password counter" and back to default password
CE	No	Transmitter Disable	M	ODU to power down transmitter
CF	Optional	Transmitter Enable	M	ODU to power up transmitter (SSPA remains off)
D0	No	Get calibration table	O	Get the ODU calibration data for temperature and frequency variations
D1	No	Get Temperature	O	Report temperature of the ODU
D2	No	Get power output value	O	Get the ODU measured power output
D3	No	Get Location	O	Get the ODU geographical location (latitude, longitude, altitude)
D4	Optional	Set Location	O	Set the ODU location (when stored in ODU)
D5	No	Serial Number	M	Request the ODU serial number
D6	No	Firmware version	M	Request the ODU firmware version
D7	No	Set Rx_Freq.	O	Set Rx Carrier Frequency to ODU
D8	No	Set Beacon_Freq	O	Set Beacon Frequency to ODU
D9	No	Set Tx_Freq	O	Set Tx Carrier Frequency to ODU
DA	No	Set Satellite_ID	O	Set Satellite ID to ODU
DB	No	Track OFF	O	Report Tracking status(OFF) to IDU
DC	No	Track ON	O	Report Tracking status(ON) to IDU
DD to DE	-	-	-	Reserved for future standard commands
DF	-	-	-	Reserved for ODU manufacturer dependent commands
NOTE: M = Mandatory, O = Optional (in case function is not supported).				

G.5.4 Password (optional)

A password may be required by the ODU for some commands in order to avoid inadvertent transmission or unapproved use of the ODU. The password will consist of 4 bytes and may be required together with the designated commands shown in Table G.3. In these cases the password will immediately follow the relevant command byte, if there is any associated data used by these commands, it will be sent in a following block (see clause G.5.5).

The ODU should refuse commands if the password does not correspond to its current valid password. After 5 consecutive erroneous passwords, the ODU should warn that only 1 try remains. After the 6th faulty password, ODU should refuse all commands except the status request, transmitter disable command and the download commands. These last commands allow an expert to reset the "Faulty password counter" and reset to the default password. The default password is system dependent.

When the ODU becomes locked due to 6 consecutive incorrect passwords, the SSPA should be disabled and the transmitter powered down.

In clause G.5.5, the password is noted "PWD" in the commands description.

G.5.5 Extended message format

In order to maintain backward compatibility with existing DiSEqC™ processors (typically 8 bit microprocessors) it is not possible to have more than 8 bytes of continuous code without the risk of potentially crashing existing devices. Therefore, to allow for the transmission of much longer messages, these will be subdivided into blocks of 8 bytes. Between each block there should be a short pause (T_b) of between 5 ms and 10 ms to allow existing microprocessors, and systems with small hardware buffers, to process each block without a data overflow.

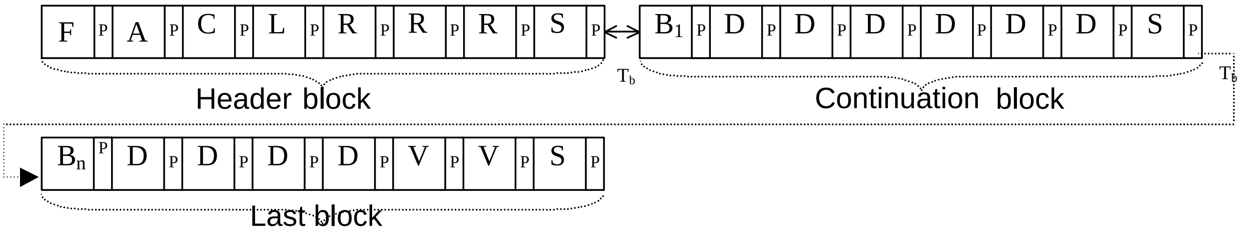
G.5.5.1 Extended messages for commands (IDU → ODU)

The structure of the first block will always be a standard DiSEqC™ message which has a framing, address and command byte, and does NOT contain any of the subsequent data which is to be error protected (e.g. CRC verified). This block will identify that the subsequent blocks are mostly data and will have a different structure, namely the first byte will be a block identifier which increments in each block, and the last byte will again be reserved for error protection. The framing byte (0x/E2/E8/EA) of the first block defines whether a reply is required to THIS initial block (before the data is transmitted), only after the last block or to all blocks. Also within the first block it will be possible to define how many blocks there are in total. An advantage of the optional reply here is that the slave can be given some time to "prepare" itself for the main data processing task (e.g. clearing a block of memory), and could delay the reply for (say) up to 100 ms, if it needed to (assuming the master has asked for a reply). If not all the subsequent blocks are to be replied to, then the LAST block could then have a reply of the form "E4" (OK), or "ED nn [nn]" (Please repeat block number[s] nn).

All subsequent (continuation) blocks would be of the form: "Ax dd dd dd dd dd dd [pp]" where A is A, B or C indicating the high nibble of the block count, x is the low nibble of the block count, d are data nibbles and pp is a simple (optional) checksum of the 6 bytes in the block. "A0" will be reserved as a "wildcard" block number for applications where it is unnecessary to update the block identifier byte for each block.

The last block contains data (or "stuffed" bytes if appropriate) AND the 16-bit CRC. The reason for this is mainly that the CRC is processed in exactly the same way as the data bits, and then if the result is 0000 the data is valid. In this way, with 6 bytes per data block, this fits 256 data bytes (+ 2 CRC bytes) neatly into exactly 43 blocks (plus the initial "header" block) which would be carried in the range of 0xA1 to 0xCB.

The extended message structure is shown below:

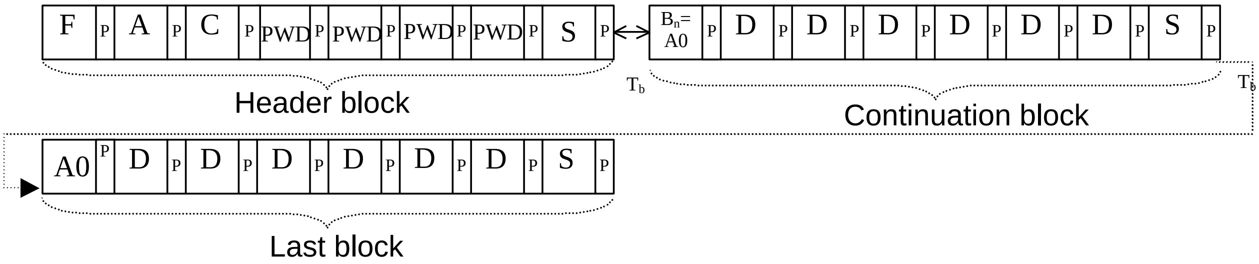

F P A P C P L P R P R P R P S P
B1 P D P D P D P D P D P D P S P
Tb
Header block
Continuation block
Tb
Bn P D P D P D P D P V P V P S P
Last block

F = Framing byte, P = Parity bit, A = Address byte, C = Command byte, L = Length of message;
R = Reserved byte (for reply strategy etc.), B_n = Block identifier, D = Data byte, S = checkSum (optional);
V = Verification (CRC as described in clause G.5.6), 5 ms < T_b < 10 ms.

Figure G.2

G.5.5.2 Simplified structure for short fixed length extended messages (IDU → ODU)

To simplify structure for short fixed messages of two or three blocks, for example password protected commands, it is possible to drop the data verification (CRC) since the likelihood of errors is much lower. As the message length (number of 8 byte blocks) is fixed and is defined by the command itself, byte "L is not required, in this case the subsequent block identifier(s) is set to "A0". To give an example for the case of a password protected command the structure could be as follows:

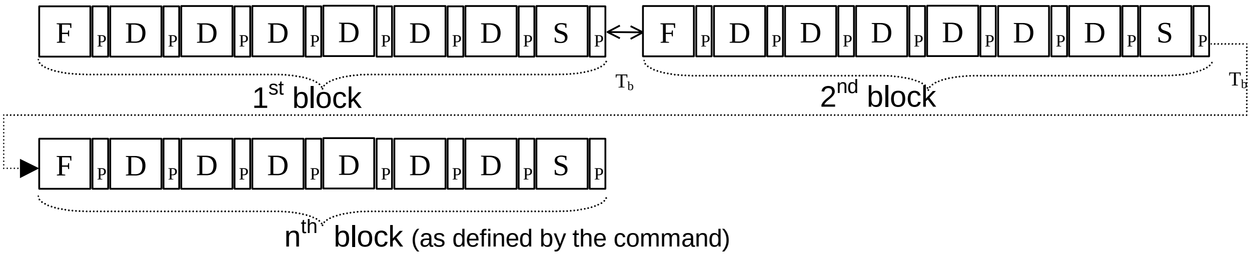

F P A P C P PWD P PWD P PWD P PWD P S P
Bn= A0 P D P D P D P D P D P D P S P
Tb
Tb
Header block
Continuation block
A0 P D P D P D P D P D P D P S P
Last block

F = Framing byte, P = Parity bit, A = Address byte, C = Command byte, PWD = Password byte, B_n = Block identifier set to A0, D = Data byte, S = checkSum (optional), 5 ms < T_b< 10 ms.

Figure G.3

G.5.5.3 Extended messages for replies (ODU → IDU)

For certain commands, the replies have additional data attached. If the total number of data bytes expected in the reply (as defined by the originating command) is more than 6 bytes then it is necessary to use the extended message structure shown below. The framing byte will usually be "E4" and the last byte is reserved for a checksum (whether it is used or not). This gives a "payload" of 6 bytes per block.


F P D P D P D P D P D P D P S P
F P D P D P D P D P D P D P S P
Tb
Tb
1st block
2nd block
F P D P D P D P D P D P D P S P
nth block (as defined by the command)

F = Framing byte, P = Parity bit, D = Data byte, S = checkSum (optional), 5 ms < T_b = 10 ms.

Figure G.4

G.5.6 CRC definition

Some commands require a CRC (see Figure G.5) at the end of the payload in order to secure the communication. The framing, destination address, command and other bytes of the first block are not included within the calculation. Only the data bytes in the subsequent blocks are processed to calculate the CRC). The CRC used is:

$$CRC = x^{16} + x^{12} + x^9 + x^5 + x + 1$$

Figure G.5: CRC calculation

G.5.7 General implementation of functions

In this clause the breakdown of each function into message exchanges between IDU and ODU is shown. These commands are not used during transmissions to avoid generating any spurious noise in the ODU.

G.5.7.1 Reset status and parameter request

G.5.7.1.1 ODU reset (0x0A) (optional)

Reset of all software ODU functions (reload PLL divider, reset register status, alarm, etc.). Note that the "Faulty password counter" will not be reset.

This command (see Table G.4) should not be sent if the SSPA is On.

Table G.4: ODU reset

Direction	Message	Comment
IDU → ODU	E2 82 0A	IDU sends reset command to the ODU.
ODU → IDU	E5	Command rejected, not supported by ODU (optional function)
ODU → IDU	E6	Command rejected, parity error during transmission.
ODU → IDU	E7	Command rejected, message format not recognized.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage).
ODU → IDU	E4	Command accepted. ODU will perform a complete reset (software reset). Faulty_passwd_counter will not be reset.

Note that "reset command" will be rejected if the ODU is locked. In fact the only way to download new software is to perform an ODU hard reset (cycling power). The IDU should wait at least 10 s after an "ODU reset" to send any command (ODU loader boot time).

G.5.7.1.2 ODU Status (0x12)

This command requests the ODU status (see Table G.5). The ODU returns the general status information to the IDU. The ODU should reply to this command even if it is in locked state. Means that the 0xED answer is not possible to this command. Alarms are buffered until the IDU reads the status register or until the IDU performs an ODU reset (0x0A).

Table G.5: ODU status

Direction	Message	Comment
IDU → ODU	E2 82 12	IDU sends status command to the ODU.
ODU → IDU	E5	Command rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Command rejected, parity error during transmission.
ODU → IDU	E7	Command rejected, message format not recognized.
ODU → IDU	E4 aa bb cc	Command accepted. ODU will give its status with 3 bytes (aa bb cc).

When the Status request command is launched while the ODU is in download mode, only "Software Download alarm" and "ODU main" fields are relevant.

G.5.7.1.2.1 aa byte status description: Alarms

Table G.6: Alarms (ODU status)

bit	Status name	Values
7	Self test alarm	0 : No Self test alarm 1 : Self test alarm
6	PLL status	0 : Lock PLL 1: Unlock PLL
5	Power supply status	0 : No Power supply Alarm 1 : Power supply Alarm
4	Faulty password counter	[0.. 6] faulty password(s) (bit 4 is msb)
3		
2		
1	Software Download alarm	0 : No CRC alarm on downloaded file 1 : CRC (see note) alarm on downloaded file
0		0 : No other download error 1 : Other download error

NOTE: This CRC corresponds to the CRC of the whole downloaded program (it does not refer to the CRC performed on each data packet - refer to 0xC2 command).

G.5.7.1.2.2 bb byte status description: ODU state

Table G.7: State (ODU status)

Bit	Status name	Values
7	Reserved	0
6	Reserved	0
5	Reserved	0
4	Reserved	0
3	SSPA Status	0 : Off 1 : On (see note)
2	ODU main	0 : Not in Running state 1 : Running state
1		Reserved : 0
0		0 : Not in Software download state 1 : Software download state

NOTE: This CRC corresponds to the CRC of the whole downloaded program (it does not refer to the CRC performed on each data packet - refer to 0xC2 command).

G.5.7.1.2.3 cc byte status description: Reserved for future use

Reserved, all bits set to zero.

G.5.7.1.3 ODU Identification (0x54, 0x55, 0x56, 0xD5)

These commands (see Table G.8) allow the factory or an authorized installer to collect the different ODU product information: ODU manufacturer's information (using EUI-64 standard from IEEE [i.5]), ODU software & hardware version/release, ODU type and ODU serial number, etc.

After power up or reset, the IDU needs to issue this command to the ODU to move to on-line mode. The IDU should wait at least 10 s after an ODU power cycling to send this command (ODU boot time).

Table G.8: ODU manufacturer's identification (0x5C)

Direction	Message	Comment
IDU → ODU	E2 82 5C	IDU sends Manufacturer's identification command to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	E4 gg gg gg	Request accepted. ODU should return the Manufacturer's OUI-24, first three bytes of EUI-64.

Table G.9: ODU product identification (0x5D)

Direction	Message	Comment
IDU → ODU	E2 82 5D	IDU sends Product identification command to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	E4 hh hh hh hh hh	Request accepted. ODU should return the Product ID, remaining 5 bytes of EUI-64.

Table G.10: ODU firmware version (0xD6)

Direction	Message	Comment
IDU → ODU	E2 82 D6	IDU sends firmware version command to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	E4 aa bb cc dd ff	Request accepted. ODU should return the ODU firmware version.

Table G.11: ODU serial number (0xD5)

Direction	Message	Comment
IDU → ODU	E2 82 D5	IDU sends serial number command to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	EF ee ee ee ee ee ee CS EF ee ee ee ee ee ee CS E4 ee ee ee ee ee ee CS	Request accepted. ODU should return the ODU serial number.

NOTE: CS = Check Sum

All the following values are considered hexadecimally coded.

Table G.12: ODU identification codes

Bytes	Bits	Status name	Values
aa	7..4	Current Software Major version	0..F
	3..0	Current Software Minor version	0..F
bb	7..4	Backup Software Major version	0..F
	3..0	Backup Software Minor version	0..F
cc	7..4	Hardware Major version	0..F
	3..0	Hardware Minor version	0..F
dd	7..3	Reserved	0
	0..2	ODU type	Gives the ODU type (1 to 4, depending on transmit symbol rate).
ee ee ee ee ee ee ee ee ee ee ee ee ee ee ee ee	127..0	ODU Serial Number	0..FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
ff	7..0	ODU Boot Firmware version	0..F
gg gg gg	63 .. 40	Company ID of the Manufacturer allocated by IEEE (OUI-24), [i.5]	Identifies Manufacturer
hh hh hh hh hh	39 .. 0	Unique Product ID allocated by Manufacturer according to [i.5]	0 .. FF FF FF FF FF

G.5.7.2 Operational commands

G.5.7.2.1 SSPA ON (0xC6)

This command forces the ODU to enable its amplification output.

Table G.13: SSPA on

Direction	Message	Comment
IDU → ODU	E2 82 C6 PWD	IDU sends the SSPA output enabling command to the ODU, for ODU requiring use of password
IDU → ODU	E2 82 C6	IDU sends the SSPA output enabling command to the ODU, for ODU not requiring use of password
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	ED Fn	Request rejected, password used not valid (< 5 faulty passwords used).
ODU → IDU	ED FE	Request rejected, 5 consecutive faulty passwords used.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	E4	Command accepted. ODU should turn on the SSPA.

G.5.7.2.2 SSPA OFF (0xC7)

This command forces the ODU to disable its amplification output.

Table G.14: SSPA off

Direction	Message	Comment
IDU → ODU	E2 82 C7	IDU sends the SSPA output disabling command to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage).
ODU → IDU	E4	Command accepted. ODU should turn off the SSPA.

By default, after a power on or a reset, the SSPA should be turned off by the ODU.

G.5.7.2.3 Transmitter disable (0xCE)

This command forces the ODU to power down the transmitter circuitry. This command will be effectuated by the ODU also when it has locked itself due to repeated use of incorrect password.

Table G.15: Transmitter Disable

Direction	Message	Comment
IDU → ODU	E2 82 CE	IDU sends the transmitter disable command to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	E4	Command accepted. ODU should disable the transmitter.

This command is issued by the IDU whenever the RCST is put in Hold State. The transmitter should not be re-enabled by the IDU as long as the RCST is in Hold State. In case of error (including internal fault conditions such as PLL unlock and/or DC powering problem) or alarm, the ODU should automatically disable the transmitter; the ODU should be unconditionally stable.

G.5.7.2.4 Transmitter enable (0xCF)

This command allows the IDU to re-enable the transmitter, e.g. when the RCST Hold State is removed. At the completion of this command, the transmitter is again powered on, but the transmitter is still in the off state.

Table G.16: Transmitter Enable

Direction	Message	Comment
IDU → ODU	E2 82 CF PWD	IDU sends the transmitter enable command to the ODU for ODUs requiring use of password.
IDU → ODU	E2 82 CF	IDU sends the transmitter enable command to the ODU for ODUs not requiring use of password.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	EA	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	E4	Command accepted. ODU should power on transmitter.

G.5.7.2.5 Set Power level (0xC8) (optional)

This command allows the IDU to adjust the output power level of the ODU in at least 1 dB or less steps. The command instructs the ODU by indicating how many "steps" up or down encoded by one signed data byte PWR_ADJ (±128 steps), this byte will follow in a separate block.

Table G.17: Set Power Level

Direction	Message	Comment
IDU → ODU	E2 82 C8 PWD PWR_ADJ	IDU sends the Set Power Level command to the ODU followed by the value in the PWR_ADJ byte, for ODUs requiring use of password.
IDU → ODU	E2 82 C8 PWR_ADJ	IDU sends the Set Power Level command to the ODU followed by the value in the PWR_ADJ byte, for ODUs not requiring use of password.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	EA	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	E4	Command accepted. ODU should change power level.

G.5.7.2.6 Mod ON (0xC9) (optional)

In the case when the modulation is applied within the ODU and a co-axial IFL is still used, then this command allows the ODU to re-enable the modulation (for future implementations).

Table G.18: Modulation On

Direction	Message	Comment
IDU → ODU	E2 82 C9	IDU sends the Mod On command to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	EA	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	E4	Command accepted. ODU should modulation on.

G.5.7.2.7 Mod OFF (0xCA) (optional)

In the case when the modulation is applied within the ODU and a co-axial IFL is still used, then this command allows the ODU to disable the modulation (for future implementations).

Table G.19: Modulation Off

Direction	Message	Comment
IDU → ODU	E2 82 CA PWD	IDU sends the Mod OFF command to the ODU for ODUs requiring use of passwords.
IDU → ODU	E2 82 CA	IDU sends the Mod OFF command to the ODU for ODUs not requiring use of passwords.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	EA	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	E4	Command accepted. ODU should switch modulation off.

When the modulation is switched off the ODU will transmit a "continuous wave" i.e. a clean carrier.

G.5.7.2.8 Set Rx Freq(0xD7) (optional)

This command allows the IDU to set Rx carrier frequency in the ODU for the ODU to track the satellite used for certain service when the ODU is (re-)initialized. This command can be used optionally when the IDU/ODU are operated in moving environment. At the completion of the command, the ODU is locked to the specified satellite and ready to receive the FLS.

Table G.20: Set Rx Freq

Direction	Message	Comment
IDU → ODU	E2 82 D7 aa aa aa aa	IDU sends the Set Rx Freq command to the ODU with 4bytes of frequency value in MHz.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	E4	Command accepted. ODU should track the satellite.

G.5.7.2.9 Set Beacon Freq(0xD8) (optional)

This command allows the IDU to set Beacon frequency in the ODU for the ODU to track the satellite used for certain service when the ODU is (re-)initialized. This command can be used optionally when the IDU/ODU are operated in moving environment. At the completion of the command, the ODU is locked to the specified satellite and ready to receive the FLS.

Table G.21: Set Beacon Freq

Direction	Message	Comment
IDU → ODU	E2 82 D8 aa aa aa aa	IDU sends the Set Beacon Freq command to the ODU with 4bytes of frequency value in MHz.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	E4	Command accepted. ODU should track the satellite.

G.5.7.2.10 Set Tx Freq(0xD9) (optional)

This command allows the IDU to set Tx carrier frequency in the ODU for the ODU to transmit the return link signal. This command can be used optionally when the IDU/ODU are operated in moving environment. And, Tx carrier frequency can be obtained from the FLS (e.g. superframe centre frequency in SCT). At the completion of the command, the ODU is ready to send user data via return-link.

Table G.22: Set Tx Freq

Direction	Message	Comment
IDU → ODU	E2 82 D9 aa aa aa aa	IDU sends the Set Tx Freq command to the ODU with 4bytes of frequency value in MHz.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	E4	Command accepted.

G.5.7.2.11 Set Satellite_ID(0xDA) (optional)

This command allows the IDU to set Satellite ID to the ODU so that the ODU can select target satellite among the several searched satellites. This command can be used optionally when the IDU/ODU are operated in moving environment. This command may be used on the premise that ODU has all satellite information such as satellite position, channel configuration, and so on. This command has to be sent by IDU in the initial step of the ODU if mobile antenna is used.

Table G.23: Set Satellite_ID

Direction	Message	Comment
IDU → ODU	E2 82 DA aa aa aa	IDU sends the Set Satellite_ID command to the ODU with 3bytes of ID value(satellite_ID : 2bytes, beam_ID : 1byte).
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	E4	Command accepted.

G.5.7.2.12 Track OFF(0xDB) (optional)

This command has to be issued by ODU whenever ODU detects missing of the satellite. This command allows the IDU to stop sending user data and start buffering. IDU can only resume sending data when it receives Track ON command from ODU.

Table G.24: Track OFF

Direction	Message	Comment
ODU → IDU	F0 81 DB	ODU sends the Track OFF command to indicate its tracking status(OFF).
IDU → ODU	F1	Command accepted.
IDU → ODU	F2	Request rejected, error during transmission.

G.5.7.2.13 Track ON(0xDC) (optional)

This command has to be issued by ODU whenever ODU re-acquires tracking of the satellite. This command allows the IDU to resume sending user data.

Table G.25: Track ON

Direction	Message	Comment
ODU → IDU	F0 81 DC	ODU sends the Track ON command to indicate its tracking status(ON).
IDU → ODU	F1	Command accepted.
IDU → ODU	F2	Request rejected, error during transmission.

G.5.7.3 Download commands

G.5.7.3.1 Download start (0xC1) (optional)

This command allows the ODU to enter into download mode. This command can only be issued after and ODU power cycle and prior to identification status request. The IDU should wait at least 10 s after an ODU power cycling to send this command (ODU loader boot time). The DL_FL_SIZE corresponds to the Download File Size expressed in bytes on 24 bits.

Table G.26: Download start

Direction	Message	Comment
IDU → ODU	E2 82 C1 DL_FL_SIZE	IDU sends the Download start command to the ODU. It includes the number of bytes of the complete software to download and the CRC on the file size.
ODU → IDU	E5	Request rejected, not supported by ODU (this answer may occur if the download start command is sent out of the allowed time after the ODU power ON event).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	E4	Command accepted. ODU should enter into download mode immediately and store the download file size.
	EC nn	Command accepted. ODU should enter into download mode, please check status after nn seconds (1 to 127 binary) and store the download file size.

Note that the DL_FL_SIZE may handle a value of 0 (zero).

If the password given is the default one, the download start command should be refused except if the ODU is locked. In this case, the ODU should enter the download mode so that a new software version can be loaded to clear the faulty password counter. This command can take up to 6,5 s to execute. IDU timeouts should account for this delay.

G.5.7.3.2 Download data (0xC2) (optional)

This command allows the IDU to transfer ODU program bytes to the ODU divided in 256 bytes per command (message) in 43 blocks of 8 bytes. If the program code is longer than 256 bytes than multiple messages each starting with 0xC2 will be used.

Table G.27: Download data

Direction	Message	Comment
IDU → ODU	E2 82 C2 L 248 data bytes	IDU sends the length of message in terms of 6 byte block of data, up to 258 bytes including 2 byte CRC - i.e. max. number of data blocks is 43. Alternative framing byte (E8 or EA) in this first block will indicate the exact reply strategy implemented.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	E4	Command accepted (first block OK) continue with data download.
IDU → ODU	Block identifier + 6 data bytes + CheckSum	L × blocks of 8 bytes (see clause G.5.5).
ODU → IDU	E4	Command accepted. ODU should store the checked data until the download validation.

Any failed packet should be ignored by the ODU.

The complete program should be stored into not sensitive memory until the validation of the complete downloaded software.

The IDU timeout period should be increased to at least 500 ms to allow for complete ODU processing of the command prior to sending the next message.

G.5.7.3.3 Download abort (0xC3) (optional)

This command allows the IDU to abort the downloading process when communication problems have occurred or on major trouble.

Table G.28: Download abort

Direction	Message	Comment
IDU → ODU	E2 82 C3	IDU sends the Download abort command to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (should only occur if software downloading is not supported).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	E4	Command accepted. ODU should remove the previous downloaded data bytes and exit the download mode in order to restore the normal running mode.

Once the abort command has been acknowledged, the IDU has to perform an ODU reset (reset or power cycling). The current software should still be active. The IDU timeout for this message should be increased to 3,5 s.

G.5.7.3.4 Download validate (0xC4) (optional)

This command allows the ODU to check the received software and store it if the received data is correct. This command may be shown as indicating the end of the download procedure.

Table G.29: Download validate

Direction	Message	Comment
IDU → ODU	E2 82 C4	IDU sends the Download validate command to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, if the new software is not valid (wrong CRC file) or the ODU is not able to store the new downloaded software or parity error during transmission of the latter command.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	E4	Command accepted. ODU should check the complete program validity and store it in order to activate this new software as the current software before acknowledging the command - if completed within 115 ms.
ODU → IDU	EC nn	Command accepted. ODU should check the complete program validity and store it in order to activate this new software as the current software before acknowledging the command, please check status after nn seconds (1 to 127 binary) if validation is complete (e.g. IDU resends 0xC4 command until E4 is received).

Before responding positively the command, the ODU should:

- Check the new software validity (CRC).
- Save the current software into the backup section.
- Save the new received software into the current software section.
- Restore the running bit into the main ODU status field.

The new program should be active only after a reset command or a power off and on. The timeout for the ODU response should be increased to as much as 8 s to accommodate required processing. The SW version will be updated in the ODU status register once the reset has been launch by the IDU. The IDU has to check the ODU status and ODU identification registers to be aware of the software download result.

G.5.7.3.5 Download toggle (0xC5) (optional)

This command allows the IDU to toggle to the previous software. This command should be send only if the ODU is in download mode. To do so, the IDU should use the "start download" command with a DL_FL_SIZE set to 0. The current software should be transfer to the backup non-volatile memory and the "old" program becomes the current one.

Table G.30: Download toggle

Direction	Message	Comment
IDU → ODU	E2 82 C5	IDU sends the Download revert command to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	E4	Command accepted. ODU should toggle the current software with the previous one (in the backup section) if completed within 115 ms.
ODU → IDU	EC nn	Command accepted. ODU should toggle the current software with the previous one (in the backup section), please check status after nn seconds (1 to 127 binary) if reversion is complete (e.g. IDU resends 0xC5 command until E4 is received).

Before responding positively the command, the ODU should switch current and "old" software program. It has to be noticed that if this command is sent twice, the ODU status will not be affected. The "old" program should be active only after a reset command or a power off and on. The SW version will be updated in the ODU status register once the reset has been launch by the IDU (this status reflects the version of the effective running software). The timeout for the ODU response can be as large as 12 s to support the processing of this command.

G.5.7.4 Password commands (optional)

The procedure to change a password is divided into 2 parts: the password change command (using the current password (PWD_cur) and the new one (PWD_new)) and the password validate command. Immediately after the acknowledgement of the password validate command, the new password becomes the current valid one. If any other command or request is inserted between the 2 password commands, the password error has to be raised, increasing the "Faulty password counter". Furthermore, the password modification procedure will have to be re-initialized.

G.5.7.4.1 Change password (0xCB) (optional)

This command enables the ODU password modification. This function only changes the password but does NOT change the "current" password validity. This command required the message to be split into two blocks as shown below.

Table G.31: Change password

Direction	Message	Comment
IDU → ODU	E2 82 CB PWD PWD_new CRC	IDU sends the change password command to the ODU with the current password value in the first block. New password calculated with CRC is sent in the second block.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	ED Fn	Request rejected, password (current) not valid (< 5 faulty passwords used).
ODU → IDU	ED FE	Request rejected, 5 consecutive faulty passwords used.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	E4	Command accepted. ODU should store the new password and wait to the next password command: validate password.

The old password is still valid at this point. As already noticed, the passwords are coded on 4 bytes.

G.5.7.4.2 Validate password (0xCC) (optional)

This changes the current password to use the new password.

Table G.32: Validate password

Direction	Message	Comment
IDU → ODU	E2 82 CC PWD_new CRC	IDU sends the validate password command to the ODU in the first block. The new password calculated with the CRC is sent in the second block.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized. This reply is the one sent by the ODU if the "validate password command" is not sent immediately after the "change password command" (see note). This should never occur. The IDU is in charge of sending the right command sequence.
ODU → IDU	ED Fn	Request rejected, password (old) not valid (< 5 faulty passwords used).
ODU → IDU	ED FE	Request rejected, 5 consecutive faulty passwords used.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage, ≥ 6 faulty passwords used).
ODU → IDU	E4	Command accepted. ODU has compared the 2 new passwords and should activate the new password.

NOTE: This reply is also used if the "change password" command is sent twice consecutively, meant that 0xCA command will be followed by 0xCA command. If the "validate password" command is not sent immediately after the "change password" command, an error is generated by the ODU, and the "faulty password counter" will be incremented. The process to modify the password exits.

The new password is valid if and only if the 2 commands "change password" and "validate password" are correctly sent with the current password and the new password. If the current password in the "change password command" or the new password in the "validate password command" is not correct, the faulty password counter should be incremented. This command should be sent immediately (consecutively) after the acknowledgement of the "change password command", otherwise the "change password command" should be discarded by the ODU (the process to modify the password exits).

This command can take up to 3,5 s to execute. The IDU timeouts should account for this delay.

G.5.7.4.3 Reset ODU locked (0xCD) (optional)

This command allows authorized personnel to reset the "Faulty password counter" and reset the default password.

Table G.33: Reset password

Direction	Message	Comment
IDU → ODU	E2 CD PWD_dft	IDU sends the Default Password to the ODU which resets the faulty password counter to zero and sets PWD_cur = PWD_disable.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.

One implementation of this command could be as follows:

- ODU is delivered with both current_password and default password set to 0000. This password should be changed before the ODU will transmit.
- During installation the Hub/IDU forces installer/user to enter the first "user" password.
- Hub records this first_user_password and ODU changes default_password and the current_password to this value.
- All subsequent changes to the current_password by the user are not recorded by the hub nor do they change the default_password in ODU.

- When ODU becomes locked:
  out of band request (e.g. by telephone call to hub) for reset
  Hub authorizes IDU to send reset command "CD" using default_password, faulty password counter is reset and ODU changes current_password to and default_password to 0000 (i.e. unable to transmit)
  Hub either forces installer/user or itself to change password, new value recorded as default_password (both at hub and in ODU) and current_password.

- ODU is unlocked.

NOTE: For added security the hub can at any time change the default_password in the ODU by using the reset command.

G.5.7.5 Other functions (optional)

G.5.7.5.1 ODU calibration table (0xD0) (optional)

This request allows the IDU to retrieve the ODU calibration matrix following the frequency and temperature curve. The format of the calibration matrix will be system and ODU dependent to account for frequency differences (e.g. Ka vs. Ku-Band), temperature variations, etc. This command is optional depending upon the implementation of the ODU.

Table G.34: ODU calibration table

Direction	Message	Comment
IDU → ODU	E2 82 D0	IDU sends the calibration matrix request to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage).
ODU → IDU	E4 aa …	Request accepted. ODU should return the output power calibration matrix.

NOTE: For a manufacturer specific table of less than 7 bytes the reply can use the simple message structure.

The power calibration matrix is ODU manufacturer dependent.

G.5.7.5.2 ODU measured temperature (0xD1) (optional)

This command allows the IDU to obtain the measured temperature of the IDU.

Table G.35: Measured temperature

Direction	Message	Comment
IDU → ODU	E2 82 D1	IDU request measured temperature from the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	EA	Request rejected, ODU locked.
ODU → IDU	E4 aa	Command accepted. ODU provides internal temperature in degrees Celsius (2's complement encoded on 1 byte).

G.5.7.5.3 ODU output power level (0xD2) (optional)

This request allows the IDU to retrieve the measured output power of the ODU.

Table G.36: ODU output power level

Direction	Message	Comment
IDU → ODU	E2 82 D2	IDU sends the output power level request to the ODU.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	ED FF	Command rejected, ODU locked (due to use of faulty password in at a previous stage).
ODU → IDU	E4 bb	Request accepted. ODU should return the output power level (encoded on one byte).

The output power level coding is ODU manufacturer dependent.

G.5.7.5.4 ODU location (0xD3) (optional)

This command allows the IDU to get the location information from the ODU.

Table G.37: Get location data

Direction	Message	Comment
IDU → ODU	E2 82 D3	IDU request to get geographical location data.
ODU → IDU	E5	Request rejected, not supported by ODU (optional function).
ODU → IDU	E6	Request rejected, parity error during transmission.
ODU → IDU	E7	Request rejected, message format not recognized.
ODU → IDU	EF xx xx xx xx, yy yy CS E4 yy yy, zz zz zz zz CS	Command accepted. ODU sends back its position co-ordinates as defined below.

NOTE 1: CS = Check Sum

- x_co-ordinate: This 32 bit field defines the x co-ordinate of the RCST location in metres;
- y_co-ordinate: This 32 bit field defines the y co-ordinate of the RCST location in metres;
- z_co-ordinate: This 32 bit field defines the z co-ordinate of the RCST location in metres.

NOTE 2: The position of the satellites will be expressed as Cartesian co-ordinates x, y, z in the geodetic reference frame ITRF96 (IERS Terrestrial Reference Frame). This system coincides with the WGS84 (World Geodetic System 84) reference system at the one metre level.

NOTE 3: These 32 bit fields are encoded in the same way as the satellite position data as spfmsbf = single precision floating point value, which is a 32 bit value formatted in accordance with ANSI/IEEE 754 [i.70]. The most significant bit (i.e. the most significant bit of the exponent) is first.

G.5.7.5.5 Set ODU location (0xD4) (optional)

This command allows the IDU to send the location information to the ODU in the case it is stored in the ODU.

Table G.38: Set location data

Direction	Message	Comment
IDU → ODU	E2 82 D3 PWD A0 xx xx xx xx yy yy CS A0 yy yy zz zz zz zz CS	IDU command to set geographical location data. The format of the position coordinates is the same as for the GET command.
ODU → IDU	E5	Request rejected, not supported by ODU (should never occur)
ODU → IDU	E6	Request rejected, parity error during transmission
ODU → IDU	E7	Request rejected, message format not recognized
ODU → IDU	E4	Command accepted. ODU stores geographical location data

NOTE: CS = Check Sum.

G.5.8 Command compatibility when SSPA ON

IDU/ODU communications should not be performed during actual return channel transmissions by the RCST to avoid the introduction of spurious signals on the transmitted carrier. In addition, some commands are not available when the SSPA is powered on. The compatibility of commands with the SSPA on is shown in Table G.39.

Table G.39: Command activity when transmitting

Hex. Byte	Command	SSPA ON
00	Reset	Not compatible
0A	Soft reset	Not compatible
12	Monitoring	Compatible
5C	Manufacturer's ID	Compatible
5D	Product ID	Compatible
C1	Download start	Not Compatible
C2	Download data	Not Compatible
C3	Download abort	Not Compatible
C4	Download valid	Not Compatible
C5	Download toggle	Not Compatible
C6	SSPA ON	--
C7	SSPA OFF	Compatible
C8	Set power level	Compatible
C9	Mod ON	Compatible
CA	Mod OFF, transmit Continuous Wave (CW)	Not Compatible
CB	Change password	Not Compatible
CC	Validate password	Not Compatible
CD	Reset ODU locked	Not Compatible
CE	Transmitter Disable	Compatible
CF	Transmitter Enable	Not compatible
D0	Get calibration data	Not Compatible
D1	Get Temperature	Compatible
D2	Get power output value	Compatible
D3	Get Location	Compatible
D4	Set Location	Not Compatible
D5	Serial Number	Compatible
D6	Firmware version	Compatible

G.5.9 Use of extended message structures

For commands with more than 4 bytes of additional data sent immediately after the command it is necessary to use either the fixed extended message structure (see clause G.5.5.2) or for very long messages (e.g. software downloading) the full extended structure (see clause G.5.5.1).

For replies with more than 7 bytes of data then it is necessary to use the extended message structure (see clause G.5.5.3).

The number of data bytes and which extended message structure to use is indicated in Table G.40.

Table G.40: Data bytes and use of message structures

Commands				Reply	
Hex. Byte	Description	No of Data Bytes	Message structure	No of Reply Data Bytes	Message structure
00	Reset	0	simple	0	simple
0A	Soft reset	0	simple	0	simple
12	Status	0	simple	3	simple
5C	Manufacturer's ID	0	simple	8	simple
5D	Product ID	0	simple	5	simple
C1	Download start	3	simple	1	simple
C2	Download data	up to 256	full extended	0	simple
C3	Download abort	0	simple	0	simple
C4	Download valid	0	simple	1	simple
C5	Download toggle	0	simple	1	simple
C6	SSPA ON	4	simple	0	simple
C7	SSPA OFF	0	simple	0	simple
C8	Set power level	5	fixed extended	0	simple
C9	Mod ON	0	simple	0	simple
CA	Mod OFF, transmit Continuous Wave (CW)	4	simple	0	simple
CB	Change password	10	fixed extended	0	simple
CC	Validate password	10	fixed extended	0	simple
CD	Reset ODU locked	4	simple	0	simple
CE	Transmitter Disable	0	simple	0	simple
CF	Transmitter Enable	4	simple	0	simple
D0	Get calibration data	0	simple	< 7	simple
D1	Get Temperature	0	simple	1	simple
D2	Get power output value	0	simple	1	simple
D3	Get Location	0	simple	12	fixed extended
D4	Set Location	16	fixed extended	0	simple
D5	Serial Number	0	simple	18	fixed extended
D6	Firmware version	0	Simple	5	simple

Annex H: Bibliography

IETF RFC 4306: "Internet Key Exchange (IKEv2) Protocol".

IETF RFC 4303: "IP Encapsulating Security Payload (ESP)".

IETF RFC 4307: "Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2)".

IETF RFC 4835: "Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)".

FIPS PUB 186-2: "Digital Signature Standard (DSS)".

FIPS 186-3: "Describes keysgreater than 1024 bits".

FIPS PUB 180-2: "Secure Hash Signature Standard (SHS) ().

NOTE: Available at http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf.

ESA Study; Security for DVB-RCS at Management and Control Planes.

NOTE: Available at http://telecom.esa.int/telecom/www/object/index.cfm?fobjectid=30289.

J. G. Proakis, 4th ed. McGraw-Hill, New York, USA, 2001: "Digital Communications".

D. Divsalar and F. Pollara,: "Turbo codes for PCS applications" in. Proc. ICC, Seattle, Washington, pp. 54-59, June 18-22, 1995.

P. Moqvist and T. Aulin: "Serially concatenated continuous phase modulation with iterative decoding", IEEE Trans. Commun., vol. 49, no. 11, pp. 1901-1915, August 2002.

P. Moqvist and T. Aulin: "Trellis termination in CPM", IEEE Electronics Letters, vol. 36, no. 23, pp. 1940-1941, November 2000.

U. Mengali and M. Morelli: "Decomposition of M-ary CPM signals into PAM waveforms", IEEE Trans. Inform. Theory, vol. 41, pp. 1265-1275, September 1995.

ETSI TS 102 771: "Digital Video Broadcasting (DVB); Generic Stream Encapsulation (GSE) implementation guidelines".

Juan Cantillo: "Cross-Layer Optimization Techniques for Satellite Communications Networks", PhD Thesis, ENST, May 2008.

N. Benvenuto, R. Dinis, D. Falconer and S. Tomasin: "Single Carrier Modulation With Nonlinear Frequency Domain Equalization: An Idea Whose Time Has Come - Again," Proceedings of the IEEE, pp. 69-96, January, 2010.

History

Document history		
V1.1.1	April 2014	Publication