Final draft ETSI ES 201 980 V4.3.0 (2023-09)

ETSI STANDARD

Digital Radio Mondiale (DRM);
System Specification

Reference

RES/JTC-DRM-38

Keywords

broadcasting, digital, DRM, radio

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 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are 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 (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, 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.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This final draft ETSI Standard (ES) 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), and is now submitted for the ETSI standards Membership Approval Procedure.

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

Modal verbs terminology

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

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

Introduction

The frequency bands used for broadcasting below 30 MHz are:

- Low Frequency (LF) band: from 148,5 kHz to 283,5 kHz, in ITU Region 1 [i.3] only;
- Medium Frequency (MF) band: from 526,5 kHz to 1 606,5 kHz, in ITU Regions 1 [i.3] and 3 [i.3] and from 525 kHz to 1 705 kHz in ITU Region 2 [i.3];
- High Frequency (HF) band: a set of individual broadcasting bands in the frequency range 2,3 MHz to 27 MHz, generally available on a Worldwide basis.

These bands offer unique propagation capabilities that permit the achievement of:

- large coverage areas, whose size and location may be dependent upon the time of day, season of the year or period in the (approximately) 11 year sunspot cycle;
- portable and mobile reception with relatively little impairment caused by the environment surrounding the receiver.

There is thus a desire to continue broadcasting in these bands, perhaps especially in the case of international broadcasting where the HF bands offer the only reception possibilities which do not also involve the use of local repeater stations.

However, broadcasting services in these bands:

- use analogue techniques;
- are subject to limited quality;
- are subject to considerable interference as a result of the long-distance propagation mechanisms which prevail in this part of the frequency spectrum and the large number of users.

As a direct result of the above considerations, there is a desire to effect a transfer to digital transmission and reception techniques in order to provide the increase in quality which is needed to retain listeners who, increasingly, have a wide variety of other programme reception media possibilities, usually already offering higher quality and reliability.

In order to meet the need for a digital transmission system suitable for use in all of the bands below 30 MHz, the Digital Radio Mondiale (DRM) consortium was formed in early 1998. The DRM consortium is a non-profit making body which seeks to develop and promote the use of the DRM system worldwide. Its members include broadcasters, network providers, receiver and transmitter manufacturers and research institutes. More information is available from their website (http://www.drm.org/).

In March 2005, the DRM Consortium voted at its General Assembly to embark on extending the capability of the DRM system to provide digital radio services at higher transmission frequencies. This range includes:

- 47 MHz to 68 MHz (Band I) allocated to analogue television broadcasting;
- 65,8 MHz to 74 MHz (OIRT FM band);
- 76 MHz to 90 MHz (Japanese FM band);
- 87,5 MHz to 107,9 MHz (Band II) allocated to FM radio broadcasting;
- 174 MHz to 240 MHz (Band III) allocated to digital broadcasting.

This extension completes the family of digital standards for radio broadcasting.

1 Scope

The present document gives the specification for the Digital Radio Mondiale (DRM) system for digital transmissions in the broadcasting bands below 300 MHz.

With respect to the previous published version, the present document adds loudness metadata provision and removes certain options from coding parameters for xHE-AAC audio.

2 References

2.1 Normative references

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

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

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

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

[1] ISO/IEC 14496-3: "Information technology - Coding of audio-visual objects - Part 3: Audio".

[2] ETSI EN 300 401: "Radio Broadcasting Systems; Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers".

[3] ISO/IEC 10646: "Information technology -- Universal Coded Character Set (UCS)".

[4] ISO 639-2: "Codes for the representation of names of languages - Part 2: Alpha-3 code".

[5] ISO 3166 (all parts): "Codes for the representation of names of countries and their subdivisions".

[6] ISO/IEC 8859-1: "Information technology - 8-bit single-byte coded graphic character sets - Part 1: Latin alphabet No. 1".

[7] ETSI TS 101 968: "Digital Radio Mondiale (DRM); Data applications directory".

[8] ISO/IEC 23003-4: "Information technology - MPEG audio technologies - Part 4: Dynamic Range Control".

[9] ISO/IEC 23003-1: "Information technology - MPEG audio technologies - Part 1: MPEG Surround".

[10] ISO/IEC 23003-3: "Information technology - MPEG audio technologies - Part 3: Unified speech and audio coding".

[11] ETSI TS 126 290: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; 5G; Audio codec processing functions; Extended Adaptive Multi-Rate - Wideband (AMR-WB+) codec; Transcoding functions (3GPP TS 26.290)".

[12] IEC 62106: "Radio Data System (RDS) - VHF/FM sound broadcasting in the frequency range from 64,0 MHz to 108,0 MHz".

[13] ETSI TS 102 386: "Digital Radio Mondiale (DRM); AM signalling system (AMSS)".

[14] ETSI TS 103 176: "Digital Audio Broadcasting (DAB); Rules of implementation; Service information features".

[15] ETSI TS 102 980: "Digital Audio Broadcasting (DAB); Dynamic Label Plus (DL Plus); Application specification".

[16] ETSI TS 103 771: "Digital Radio Mondiale (DRM); Regional profiles".

[17] Unicode® standard.

[18] Unicode® bidirectional algorithm, UAX#9.

2.2 Informative references

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

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

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

[i.1] Recommendation ITU-R BS.1615: "Planning parameters" for digital sound broadcasting at frequencies below 30 MHz".

[i.2] Recommendation ITU-R BS.1660: "Technical basis for planning of terrestrial digital sound broadcasting in the VHF band".

[i.3] ITU Radio Regulations.

[i.4] Recommendation ITU-R BS.1770-4: "Algorithms to measure audio programme loudness and true-peak audio level".

3 Definition of terms, symbols, abbreviations and conventions

3.1 Terms

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

cell: sine wave portion of duration T_s, transmitted with a given amplitude and phase and corresponding to a carrier position

NOTE: Each OFDM symbol is the sum of K such sine wave portions equally spaced in frequency.

energy dispersal: operation involving deterministic selective complementing of bits in the logical frame, intended to reduce the possibility that systematic patterns result in unwanted regularity in the transmitted signal

Fast Access Channel (FAC): channel of the multiplex data stream which contains the information that is necessary to find services and begin to decode the multiplex

kbit/s: kilo bits per second (1 000 bits per second)

logical frame: data contained in one stream during 400 ms or 100 ms

Main Service Channel (MSC): channel of the multiplex data stream which occupies the major part of the transmission frame and which carries all the digital audio services, together with possible supporting and additional data services

mod: modulo operator

NOTE: (x mod y) = z, where y > 0, such that x = qy + z, q is an integer and $0 \leq z < y$.

multiplex frame: logical frames from all streams form a multiplex frame

NOTE: It is the relevant basis for coding and interleaving.

OFDM symbol: transmitted signal for that portion of time when the modulating amplitude and phase state is held constant on each of the equally-spaced carriers in the signal

reserved for future addition (rfa): bits with this designation are set to zero

NOTE: Receivers need not decode these bits.

reserved for future use (rfu): bits with this designation are set to zero

NOTE: Receivers need to check these bits in order to determine the valid status of the other fields in the same scope.

Service Description Channel (SDC): channel of the multiplex data stream which gives information to decode the services included in the multiplex

NOTE: The SDC also provides additional information to enable a receiver to find alternative sources of the same data.

Single Frequency Network (SFN): network of transmitters sharing the same radio frequency to achieve a large area coverage

transmission frame: number of consecutive OFDM symbols, whereby the first OFDM symbol contains the time reference cells

transmission super frame: set of consecutive transmission frames, whereby the first OFDM symbols contain the SDC block

UEP profile: combination of protection levels and lengths of higher protected parts for unequal error protection

3.2 Symbols

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

$E[\,]$	expectation value of the expression in brackets
f_R	reference frequency of the emitted signal
K	number of active carriers in the OFDM symbol
$K_{\max}$	carrier index of the upper active carrier in the OFDM signal
$K_{\min}$	carrier index of the lower active carrier in the OFDM signal
L_{MUX}	number of input bits per multiplex frame for the multilevel encoding
N_{MUX}	number of MSC cells (QAM symbols) per multiplex frame
T	elementary time period, equal to $83^{1/3}$ μs (1/12 kHz)
T_f	duration of the transmission frame
T_g	duration of the guard interval
T_s	duration of an OFDM symbol
T_{sf}	duration of the transmission super-frame built from the set of transmission frames
T_u	duration of the useful (orthogonal) part of an OFDM symbol, excluding the guard interval
X^*	complex conjugate of value X
$\lceil\ \rceil$	round towards plus infinity
$\lfloor\ \rfloor$	round towards minus infinity

3.3 Abbreviations

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

AAC	Advanced Audio Coding
ACELP	Algebraic Code Excited Linear Prediction
AF	Audio Frequency
AFS	Alternative Frequency Switching
AM	Amplitude Modulation
AMR-WB	Adaptive Multi-Rate WideBand
AMSS	Amplitude Modulation Signalling System
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
CA	Conditional Access
CC	Country Codes
CCIR	Comité Consultatif International des Radio-communications
CI	Continuity Index
CIRAF	Conferencia Internacional de Radiodifusión por Altas Frecuencias
C/N	Carrier to Noise
CRC	Cyclic Redundancy Check
DAB	Digital Audio Broadcasting
dBFS	deciBels Full Scale
DC	Direct Current
DL	Dynamic Label
DRM	Digital Radio Mondiale
DSB	Double SideBand
ECC	Extended Country Code
EEP	Equal Error Protection
EP	Error Protection
ER	Error Robust
EW	East West
FAC	Fast Access Channel
FEC	Forward Error Correction
FIG	Fast Information Group
FM	Frequency Modulation
HCR	Huffman Codeword Reordering
HE-AAC	High Efficiency AAC
HF	High Frequency
HFCC	High Frequency Coordination Committee
ID	Identification
IdLQ	Identifier List Qualifier
IFFT	Inverse Fast Fourier Transform
ILS	International Linkage Set
ISO	International Organization for Standardization
LA	Linkage Actuator
LF	Low Frequency
LKFS	Loudness, K-weighted, relative to nominal Full Scale
LOS	Line-Of-Sight
LRI	Left-to-Right Isolate
LSb	Least Significant bit
LSN	Linkage Set Number
LTO	Local Time Offset
LTR	Left-To-Right
MDCT	Modified Discrete Cosine Transform
MDI	Multiplex Distribution Interface
MF	Medium Frequency
MJD	Modified Julian Date
MPEG	Moving Picture Experts Group
MPS	MPEG Surround
MSb	Most Significant bit
MSC	Main Service Channel

MW	Medium Wave
NS	North South
OFDM	Orthogonal Frequency Division Multiplexing
OIRT	Organization Internationale de Radiodiffusion et de Télévision
PDI	Pop-Directional-Isolate
PDS	Power Density Spectrum
PI	Programme Identifier
PNS	Perceptual Noise Substitution
PPI	Padded Packet Indicator
PRBS	Pseudo-Random Binary Sequence
PS	Parametric Stereo
QAM	Quadrature Amplitude Modulation
QMF	Quadrature Mirror Filter
RDS	Radio Data System
RF	Radio Frequency
rfa	reserved for future addition
rfu	reserved for future use
RLI	Right-to-Left Isolate
RM	Robustness Mode
RS	Reed-Solomon
RTL	Right-To-Left
RVLC	Reversible Variable Length Coding
SAC	Spatial Audio Coding
SBR	Spectral Band Replication
SDC	Service Description Channel
SFN	Single Frequency Network
SI	Side Information
SId	Service Identifier
SNR	Signal to Noise Ratio
SSB	Single SideBand
SW	Short Wave
TCX	Transform Coded eXcitation
TES	Temporal Envelope Shaping
TNS	Temporal Noise Shaping
TSD	Transient Steering Decorrelator
UEP	Unequal Error Protection
uimsbf	unsigned integer most significant bit first
UK	United Kingdom
US	United States
USAC	Unified Speech and Audio Coding
UTC	Co-ordinated Universal Time
UTF	Unicode Transformation Format
VCB11	Virtual Codebooks for Codebook 11
VHF	Very High Frequency
VSB	Vestigial SideBand
WSSUS	Wide Sense Stationary Uncorrelated Scattering model
xHE-AAC	eXtended HE-AAC
XOR	eXclusive OR

3.4 Conventions

Unless otherwise stated, the following convention, regarding the order of bits within each step of processing is used:

- in figures, the bit shown in the left hand position is considered to be first;
- in tables, the bit shown in the left hand position is considered to be first;
- in numerical fields, the Most Significant bit (MSb) is considered to be first and denoted by the higher number. For example, the MSb of a single byte is denoted "b7" and the Least Significant bit (LSb) is denoted "b0";
- in vectors (mathematical expressions), the bit with the lowest index is considered to be first.

4 General characteristics

4.1 System overview

The DRM system is designed to be used at any frequency below 300 MHz, with variable channelization constraints and propagation conditions throughout these bands. In order to satisfy these operating constraints, different transmission modes are available. A transmission mode is defined by transmission parameters classified in two types:

- signal bandwidth related parameters;
- transmission efficiency related parameters.

The first type of parameters defines the total amount of frequency bandwidth for one transmission. Efficiency related parameters allow a trade-off between capacity (useful bit rate) and ruggedness to noise, multipath and Doppler.

4.2 System architecture

This clause gives a general presentation of the system architecture, based on the synoptic diagram of figure 1, which gives reference to the clauses defining the individual parts of the system.

Figure 1 describes the general flow of different classes of information (audio, data, etc.) and does not differentiate between different services that may be conveyed within one or more classes of information. A detailed description on the distribution of services onto those classes can be found in clause 6.

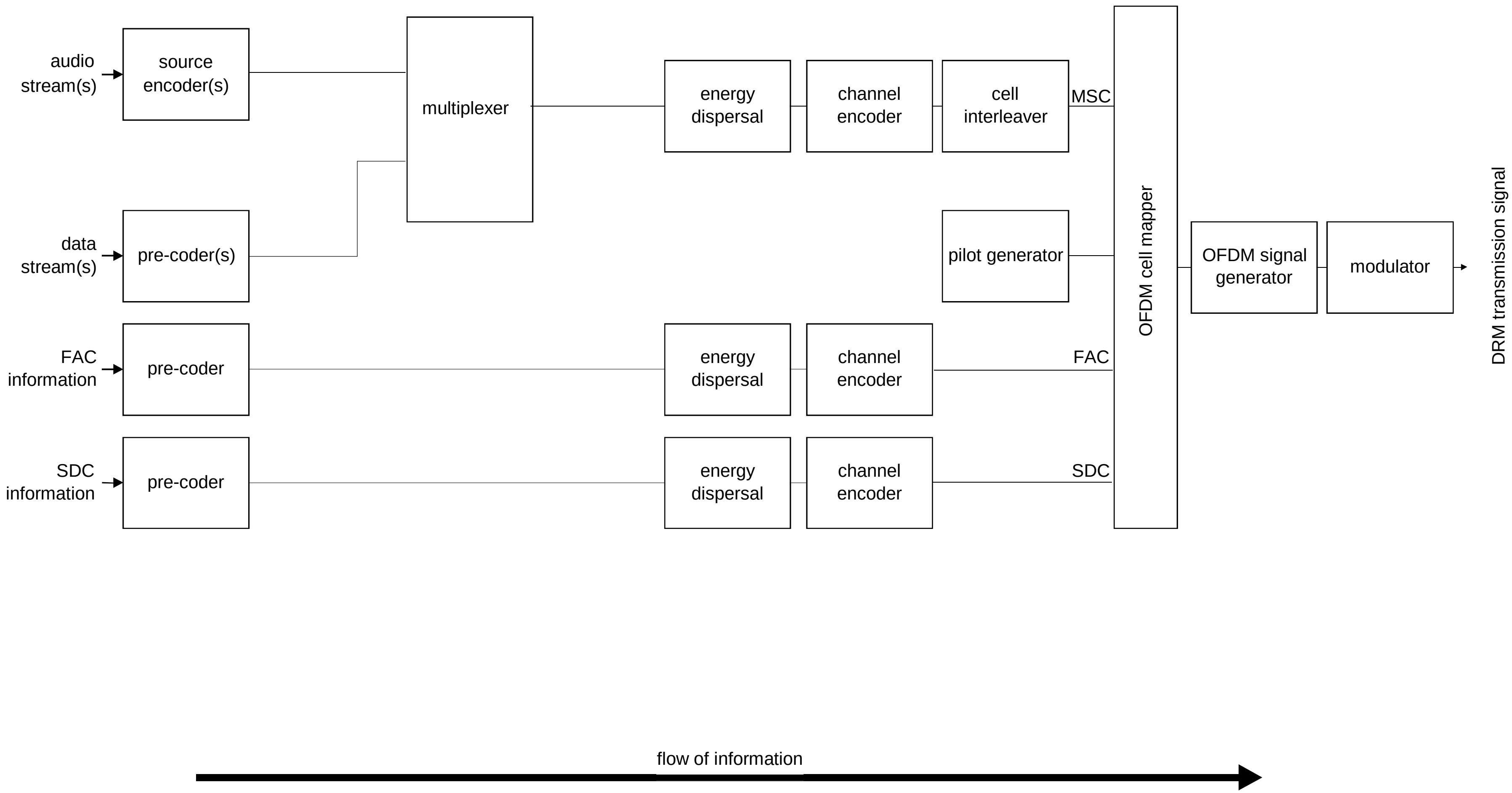


Figure 1: Conceptual DRM transmission block diagram

The source encoder(s) and pre-coders ensure the adaptation of the input streams onto an appropriate digital transmission format. For the case of audio source encoding, this functionality includes audio compression techniques as described in clauses 4.3 and 5. The output of the source encoder(s) and the data stream pre-coder(s) may comprise data for one or two different levels of protection within the subsequent channel encoder. All services have to choose from the same two levels of protection.

The multiplexer combines the protection levels of all data and audio services as described in clause 6.

The energy dispersal provides a deterministic selective complementing of bits in order to reduce the possibility that systematic patterns result in unwanted regularity in the transmitted signal.

The channel encoder adds redundant information as a means for quasi error-free transmission and defines the mapping of the digital encoded information onto QAM cells as described in clause 7.

Cell interleaving spreads consecutive QAM cells onto a sequence of cells quasi-randomly separated in time and frequency, in order to provide robust transmission in time-frequency dispersive channels. The pilot generator provides means to derive channel state information in the receiver, allowing for a coherent demodulation of the signal.

The OFDM cell mapper collects the different classes of cells and places them on the time-frequency grid as specified in clause 7.

The OFDM signal generator transforms each ensemble of cells with same time index to a time domain representation of the signal. Consecutively, the OFDM symbol is obtained from this time domain representation by inserting a guard interval as a cyclic repetition of a portion of the signal, as specified in clause 7.

The modulator converts the digital representation of the OFDM signal into the analogue signal in the air. This operation involves digital-to-analogue conversion and filtering.

4.3 Audio source coding

Within the constraints of broadcasting regulations in broadcasting channels below 30 MHz and the parameters of the coding and modulation scheme applied, the bit rate available for source coding is in the range from 8 kbit/s (half channels) to ≈20 kbit/s (standard channels) to up to ≈72 kbit/s (double channels).

Within the constraints of broadcasting regulations in broadcasting channels above 30 MHz and the parameters of the coding and modulation scheme applied, the bit rate available for source coding is in the range from 37 kbit/s to 186 kbit/s.

The system offers different audio source coding schemes:

- a subset of MPEG xHE-AAC (Extended High-Efficiency Advanced Audio Coding) for mono and stereo audio broadcasting;
- a subset of MPEG-4 AAC (Advanced Audio Coding) including error robustness tools for mono and stereo audio broadcasting, including:
    - Spectral Band Replication (SBR), an audio coding enhancement tool that allows the full audio bandwidth to be achieved at low bit rates;
    - Parametric Stereo (PS), an audio coding enhancement tool relevant to SBR that allows for stereo coding at low bit rates;
- MPEG Surround (MPS), an audio coding enhancement tool that allows for multichannel coding at low bit rates.

For typical broadcast scenarios the use of xHE-AAC audio coding will achieve a better audio quality and therefore should be preferred over AAC audio coding.

Provision is made for further enhancement of the audio system by linking two DRM signals together.

4.4 Transmission modes

4.4.1 Signal bandwidth related parameters

The current channel widths for radio broadcasting below 30 MHz are 9 kHz and 10 kHz. The DRM system is designed to be used:

- within these nominal bandwidths, in order to satisfy the current planning situation;
- within half these bandwidths (4,5 kHz or 5 kHz) in order to allow for simulcast with analogue AM signals;
- within twice these bandwidths (18 kHz or 20 kHz) to provide for larger transmission capacity where and when the planning constraints allow for such facility.

The current channel raster (where defined) for analogue radio broadcasting above 30 MHz is 100 kHz. The DRM system is designed to be used with this raster.

These signal bandwidth related parameters are specified in clause 8.

4.4.2 Transmission efficiency related parameters

4.4.2.0 General

For any value of the signal bandwidth parameter, transmission efficiency related parameters are defined to allow a trade off between capacity (useful bit rate) and ruggedness to noise, multipath and Doppler. These parameters are of two types:

- coding rate and constellation parameters, defining which code rate and constellations are used to convey data;
- OFDM symbol parameters, defining the structure of the OFDM symbols to be used as a function of the propagation conditions.

4.4.2.1 Coding rates and constellations

As a function of the desired protection associated within each service or part of a service, the system provides a range of options to achieve one or two levels of protection at a time. Depending on service requirements, these levels of protection may be determined by either the code rate of the channel encoder (e.g. 0,6, etc.), or by the constellation order (e.g. 4-QAM, 16-QAM, 64-QAM). Detailed definition of these options is given in clause 7.

4.4.2.2 OFDM parameter set

The OFDM parameter set is presented in this clause. The specification of the signal waveform is given in clause 8. These values are defined for different propagation-related transmission conditions to provide various robustness modes for the signal. In a given bandwidth, the different robustness modes provide different available data rates. Table 1 illustrates typical uses of the robustness modes.

Table 1: Robustness mode uses

Robustness mode	Typical propagation conditions
A	Gaussian channels, with minor fading
B	Time and frequency selective channels, with longer delay spread
C	As robustness mode B, but with higher Doppler spread
D	As robustness mode B, but with severe delay and Doppler spread
E	Time and frequency selective channels

The transmitted signal comprises a succession of OFDM symbols, each symbol being made of a guard interval followed by the so-called useful part of the symbol. Each symbol is the sum of K sine wave portions equally spaced in frequency. Each sine wave portion, called a "cell", is transmitted with a given amplitude and phase and corresponds to a carrier position. Each carrier is referenced by the index *k*, *k* belonging to the interval $[k_{\min}, k_{\max}]$ ($k = 0$ corresponds to the reference frequency of the transmitted signal).

The time-related OFDM symbol parameters are expressed in multiples of the elementary time period *T*, which is equal to 83⅓ µs. These parameters are:

- T_g: duration of the guard interval;
- T_s: duration of an OFDM symbol;
- T_u: duration of the useful (orthogonal) part of an OFDM symbol (i.e. excluding the guard interval).

The OFDM symbols are grouped to form transmission frames of duration T_f.

As specified in clause 8, a certain number of cells in each OFDM symbol are transmitted with a predetermined amplitude and phase, in order to be used as references in the demodulation process. They are called "reference pilots" and represent a certain proportion of the total number of cells.

Table 2: OFDM symbol parameters

Parameters list	Robustness mode				
	A	B	C	D	E
T (µs)	83⅓	83⅓	83⅓	83⅓	83⅓
T_u (ms)	24 (288 × T)	21⅓ (256 × T)	14⅔ (176 × T)	9⅓ (112 × T)	2¼ (27 × T)
T_g (ms)	2⅔ (32 × T)	5⅓ (64 × T)	5⅓ (64 × T)	7⅓ (88 × T)	0¼ (3 × T)
T_g / T_u	1/9	1/4	4/11	11/14	1/9
$T_s = T_u + T_g$ (ms)	26⅔	26⅔	20	16⅔	2½
T_f (ms)	400	400	400	400	100

5 Source coding modes

5.1 Overview

5.1.0 Introduction

The audio source coding options in the DRM system are shown in figure 2. As described in clause 4.3, the DRM system offers two audio codecs, xHE-AAC and AAC (in combination with the SBR and PS tools). Optionally, MPS can be used to enable multichannel coding. The encoded audio is composed into audio super frames of constant length. Multiplexing of audio services is done by means of the multiplex and channel coding units. Audio specific configuration information is transmitted in the SDC (see clause 6.4.3.10).

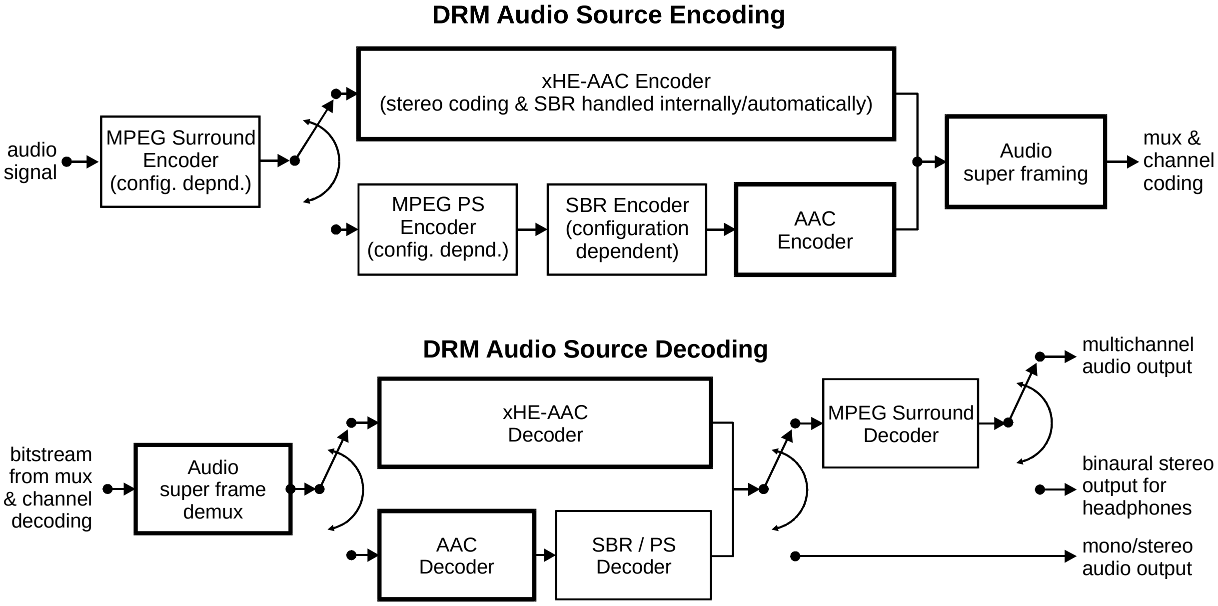


Figure 2: Audio source coding overview

5.1.1 Extended HE-AAC audio coding (xHE-AAC)

For generic coding of both audio and speech content at all bit rates, a subset of the MPEG xHE-AAC toolbox chosen to best suit the DRM system environment is used. For example a standard configuration for use in one short wave channel could be 16 kbit/s stereo.

Specific features of the xHE-AAC stream within the DRM system are:

- Bit rate: xHE-AAC can be used at any bit rate. The granularity of the xHE-AAC bit rate is 20 bit/s for robustness modes A, B, C and D and 80 bit/s for robustness mode E.
- Sampling rates: Permitted sampling rates for the use of xHE-AAC within DRM are selected such that the interface of the xHE-AAC codec towards the surrounding application can easily accept or provide 48 kHz audio signals, respectively. The actual core sampling rate is selected by the encoder upon initialization to ensure the best possible audio signal quality and is typically not visible to higher layers of processing.
- Audio super framing: To ensure the best possible audio quality particularly at lower bit rates, the xHE-AAC encoder can flexibly assign the available bit rate within certain constraints to each audio frame. Audio super frames - as generated by the xHE-AAC audio encoder and inserted into the DRM logical frames - always have a constant size. However, the number of audio frames per audio super frame is not fixed, and audio frames may span audio super frames. This flexibility is achieved by a slight adjustment to the audio super frame header configuration used for the AAC codec in DRM. One audio super frame is always placed in one DRM logical frame in robustness modes A, B, C and D and in two logical frames in robustness mode E (see clause 6). In this way no additional synchronization is needed for the audio coding. Retrieval of frame boundaries is also taken care of within the audio super frame.

5.1.2 AAC audio coding

For generic audio coding, a subset of the MPEG-4 Advanced Audio Coding (AAC) toolbox chosen to best suit the DRM system environment is used. For example a standard configuration for use in one short wave channel could be 20 kbit/s mono.

Specific features of the AAC stream within the DRM system are:

- Bit rate: AAC can be used at any bit rate. The granularity of the AAC bit rate is 20 bit/s for robustness modes A, B, C and D and 80 bit/s for robustness mode E.

- Sampling rates: permitted sampling rates are 12 kHz and 24 kHz for robustness modes A, B, C and D and 24 kHz and 48 kHz for robustness mode E. 48 kHz is only permitted if the SBR tool is not used.
- Transform length: the transform length is 960 to ensure that one audio frame corresponds to 80 ms or 40 ms (robustness modes A, B, C and D) or to 40 ms or 20 ms (robustness mode E) in time. This is required to allow the combination of an integer number of audio frames to build an audio super frame of 400 ms (robustness modes A, B, C and D) or 200 ms (robustness mode E) duration.
- Error robustness: a subset of MPEG-4 tools is used to improve the AAC bit stream error robustness in error prone channels (the MPEG-4 EP tool is not used).
- Audio super framing: 5 or 10 audio frames are composed into one audio super frame. For robustness modes A, B, C and D, the respective sampling rates are 12 kHz and 24 kHz producing an audio super frame of 400 ms duration; for robustness mode E, the respective sampling rates are 24 kHz and 48 kHz producing an audio super frame of 200 ms duration. The audio frames in the audio super frames are encoded together such that each audio super frame is of constant length, i.e. that bit exchange between audio frames is only possible within an audio super frame. One audio super frame is always placed in one logical frame in robustness modes A, B, C and D and in two logical frames in robustness mode E (see clause 6). In this way no additional synchronization is needed for the audio coding. Retrieval of frame boundaries are also taken care of within the audio super frame.

SBR coding

To maintain a reasonable perceived audio quality at low bit rates, classical audio or speech source coding algorithms need to limit the audio bandwidth and to operate at low sampling rates. It is desirable to be able to offer high audio bandwidth also in very low bit rate environments. This can be realized by the use of Spectral Band Replication (SBR).

The purpose of SBR is to recreate the missing high frequency band of the audio signal that could not be coded by the encoder. In order to do this in the best possible way, some side information needs to be transmitted in the audio bitstream, removing a small percentage of the available data rate from the audio coder. This side information is computed on the full bandwidth signal, prior to encoding and aids the reconstruction of the high frequencies after audio/speech decoding.

SBR exists in two versions. The version difference is only reflected in the decoder design. High Quality SBR uses a complex filterbank whereas Low Power SBR uses a real-valued filterbank plus anti-aliasing modules. The Low Power version of SBR offers a significant reduction in complexity as compared to the High Quality version without compromising too much on audio quality. AAC + SBR is defined in MPEG-4 Audio (High Efficiency AAC profile) [1].

PS coding

For improved performance at low bit rate stereo coding, a Parametric Stereo (PS) coder is available. The PS tool can be used when running the configuration AAC + SBR (MPEG High Efficiency AAC profile). The general idea with PS coding is to send stereo image describing data as side information along with a downmixed mono signal. This stereo side information is very concise and only requires a small fraction of the total bit rate allowing the mono signal to have maximum quality for the total bit rate given.

The stereo synthesis at the decoder reconstructs spatial properties but does not affect the total spectral energy. Hence, there is no colorization of the frequency spectrum compared to the mono compatible core signal. The target bit rates for applying parametric stereo coding on AAC + SBR are preferably any bit rate range where traditional stereo cannot be afforded.

If the broadcast signal contains PS data, the PS tool as specified in MPEG-4 Audio [1] shall be used.

Error concealment

For AAC and for the SBR and PS tools a description for the concealment of erroneous bit streams is given. The error concealment provided by a DRM decoder shall provide at least the same level of performance as the specified concealment tools, but may be enhanced by specific implementations.

5.1.3 MPEG Surround coding

An MPEG Surround (MPS) coder is available for mono/stereo compatible multichannel encoding. MPEG Surround is standardized in MPEG-D, ISO/IEC 23003-1 [9]. It describes:

- coding of multichannel signals based on a downmixed signal of the original multichannel signal, and associated spatial parameters. It offers lowest possible data rate for coding of multichannel signals, as well as an inherent mono or stereo downmix signal included in the data stream. Hence, a mono or stereo signal can be expanded to multi-channel by a very small additional data overhead;
- binaural decoding of the MPEG Surround stream, enabling a surround sound experience over stereo headphones;
- an Enhanced Matrix Mode that enables a multi-channel upmix from a stereo signal without any spatial parameters.

Receivers without multichannel decoding support shall decode the unmodified mono or stereo core signal.

Hence, MPEG Surround (Spatial Audio Coding, SAC) is capable of re-creating N channels based on M<N transmitted channels, and additional control data. In the preferred modes of operating the spatial audio coding system, the M channels can either be a single mono channel or a stereo channel pair. The control data represents a significantly lower data rate than required for transmitting all N channels, making the coding very efficient while at the same time ensuring compatibility with both M channel devices and N channel devices.

The MPEG Surround standard incorporates a number of tools enabling a number of features that allow for broad application of the standard. A key feature is the ability to scale the spatial image quality gradually from very low spatial overhead towards transparency. Another key feature is that the compatible decoder input can be made compatible to existing matrix surround technologies. All tools are grouped to cover certain profiles.

Receivers with a different number of output channels than the number of MPS target channels indicated by the SDC shall render the multichannel audio signal according to the available number of output channels (possibly at a reduced quality compared to the case where the number of target channels matches the number of output channels).

5.2 Audio super framing

The channel coding of DRM is performed on logical frames of constant bit rate for any given combination of parameters. However, xHE-AAC and AAC are coding schemes with a variable length, and therefore several coded audio frames, or parts of coded audio frames, are grouped together to build one audio super frame. The bit rate of the audio super frame is constant.

Table 3: Syntax of audio_super_frame()

Syntax	No. of bits	Note

audio_super_frame(audio_info) //audio info from the SDC
{
    switch (audio_info.audio_coding) {
        case xHE-AAC:
            xhe_aac_super_frame(audio_info);
            break;
        case AAC:
            aac_super_frame(audio_info);
            break;
    }
}		

NOTE: The SDC describes the audio coder used, and the parameters associated with that coder. It also provides information about the sampling rate and bit rate used (see clause 6).		

For robustness modes A, B, C and D, the audio super frame is mapped directly onto the logical frame, since both are of the same duration.

For robustness mode E, the audio super frame is mapped onto two logical frames, since the audio super frame is of twice the duration of the logical frame.

Figure 3: Void

5.3 xHE-AAC coding

5.3.1 xHE-AAC

5.3.1.0 Introduction

The core technology of MPEG Extended HE-AAC is the MPEG standard Unified Speech and Audio Coding (USAC) defined in ISO/IEC 23003-3 [10]. USAC allows for coding of speech, audio or any mixture of speech and audio with a consistent audio quality for all sound material over a wide range of bit rates. It enables very efficient coding at very low bit rates while retaining the full audio bandwidth. In order to achieve equally good quality for coding audio and speech, USAC employs the proven MDCT-based transform coding techniques known from MPEG-4 audio and combines them with specialized speech coder elements like ACELP. Parametric coding tools such as MPEG-4 Spectral Band Replication (SBR) and MPEG-D MPEG Surround were enhanced and tightly integrated into the codec. The result delivers highly efficient coding and operates down to the lowest bit rates.

DRM specific usage of xHE-AAC:

From the possible audio object types, only the USAC object type (Audio Object Type ID = 42), which is part of the Extended High Efficiency AAC Profile and the USAC Baseline Profile, is used in the DRM system. The xHE-AAC audio codec shall comply with level 1 (mono) and 2 (stereo) of the USAC Baseline Profile. Discrete multichannel audio is not supported (instead, the MPEG Surround technology is used if required).

The following rules apply to the use of xHE-AAC in DRM:

- The MPS212 parametric stereo tool is always used with the decorrelator configuration corresponding to bsDecorrConfig==0 in ISO/IEC 23003-3 [10].
- QMF based Harmonic Transposer (harmonicSBR==1) may only be employed:
  - in mono operation; or
  - in stereo operation if parametric stereo (MPS212) without residual coding is applied, i.e. if stereoConfigIndex==1 (see clause 5.3.2).
- If temporal shaping is applied in the MPS212 parametric stereo tool, the Transient Steering Decorrelator (TSD) method shall be employed.
- Stereo operation shall be facilitated by the use of one channel pair element (ID_USAC_CPE; see clause 5.3.2, xHE-AAC codec configuration), instead of transmitting two single channel elements.
- The usacElementType ID_USAC_LFE is not used (see clause 5.3.2).
- The 4:1 sbrRatio (sbrRatioIndex==1 in ISO/IEC 23003-3 [10]) may only be employed:
  - in mono operation; or
  - in stereo operation if parametric stereo (MPS212) without residual coding is applied, i.e. if stereoConfigIndex==1 (see clause 5.3.2);
  - at sampling rates ≥ 32 kHz.
- The standard UsacExtElement() is used to carry MPEG extension payload information, as defined by ISO/IEC 23003-3 [10]. The corresponding static part of the configuration is signalled via SDC data entity type 9 in the *codec specific configuration* field.
- The maximum bit rate of an MSC stream carrying xHE-AAC content is limited to 163 920 bps.
- The xHE-AAC encoder shall not open more than 15 audio frames per audio super frame (i.e. each directory shall have a maximum of 15 entries).

- It is recommended to include loudness metadata in the xHE-AAC audio configuration. This shall be done by means of the UsacConfigExtensionDrm() structure.
- If loudness metadata is present in the UsacConfigExtensionDrm() structure, the following shall apply:
  - it shall be carried as a single instance of either the extension type ID_CONFIG_EXT_LOUDNESS_INFO_DRM or the extension type ID_CONFIG_EXT_LOUDNESS_INFO_DRM_SAMPLEPEAK (see clause 5.3.2);
  - the value carried in the loudness metadata shall:
    - be accurate and consistent in the sense that it correctly describes the long-term average loudness of the audio service;
    - only change at DRM reconfigurations;
    - be in compliance with regional broadcast union regulations for content loudness.

NOTE 1: Since the value carried in the loudness metadata can only be changed by signalling a reconfiguration, the service provider will usually set a static target loudness level. However, for a multi-genre service a broadcaster may have a different target loudness level for each genre.

NOTE 2: For regions without a common loudness regulation, broadcasters are still encouraged to normalize their programmes within a service to a common loudness value, and correctly signal it to the receivers. The selected loudness value should follow established regulations of major broadcast unions.

5.3.1.1 xHE-AAC audio super frame

Each xHE-AAC audio super frame is of constant size.

The xHE-AAC audio super frame is split into three sections:

- Header section:
  indicating the number of frame borders within this audio super frame, and the bit reservoir fill level related to the first audio frame starting within the audio super frame.
- Payload section:
  a sequence of audio frames.
  Audio frames may continue between the Payload sections of consecutive audio super frames. Audio frames are stored in the Payload section as full-byte blocks and as a continuous sequence, without any padding bytes in-between.
- Directory section:
  indicating the position of frame borders within this audio super frame.

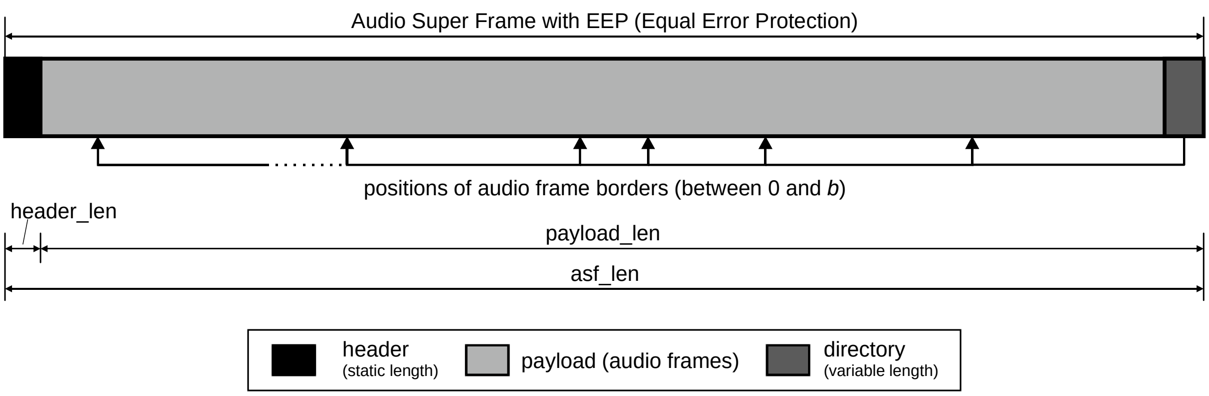


Figure 4: Example for xHE-AAC audio super frame structure

Header section

The header section indicates the total number of audio frames starting within the payload section of the current audio super frame. In addition it indicates the bit reservoir level of the first audio frame starting in the current audio super frame - the bit reservoir level values of following audio frames can be derived by the decoder.

The xHE-AAC audio super frame Header section has the following structure:

- Frame border count 4 bits.
- Bit reservoir level 4 bits.
- Fixed header CRC 8 bits.

The following definitions apply:

Frame border count: this 4-bit field contains as an integer numeric value the number *b* of *Frame border description* elements carried in the Directory section of the audio super frame. This value is equal to the number of audio frame borders contained within the Payload section of the current audio super frame, unless the *Frame border description* element related to the last audio frame border is carried in the following audio super frame due to lack of remaining capacity; see clause 5.3.1.3. *b* may be 0.

Bit reservoir level: this 4-bit field signals as an unsigned integer numeric value the fill level of the encoder's internal bit reservoir per channel after encoding the first audio frame started within the current audio super frame; if the audio super frame does not contain any frame borders, then this value represents the bit reservoir level of the audio frame that is (partly) carried in the current audio super frame. See clause 5.3.1.3.

Fixed header CRC: this 8-bit CRC value is calculated as defined in annex D over the *Frame border count* field and the *Bit reservoir level* field.

Directory section

The Directory section indicates for each audio frame starting within the Payload section of the current audio super frame a pointer to the audio frame border, i.e. the 0-based index value of the first byte of the audio frame, counted from the beginning of the Payload section.

The xHE-AAC audio super frame Directory section has the following structure:

- Frame border description $b \times 16$ bits.

The following definitions apply:

Frame border description: this field carries *b* consecutive *Frame border description* elements. The order of description elements corresponds to the order of audio frame borders in the payload section in reverse order, i.e. the first audio frame border is carried in the last *Frame border description* element. Each *Frame border description* element has a fixed length of 16 bits and the following structure:

- Frame border index 12 bits.
- Frame border count 4 bits.

Frame border index: this 12-bit field contains as an unsigned integer numeric value the 0-based index position of the first byte of the corresponding audio frame in the Payload section of the audio super frame, counted from the beginning of the Payload section. Index values 0xFFE and 0xFFF refer to a frame border location at the end of the previous audio super frame (see below).

Frame border count: this 4-bit field contains the *Frame border count* value carried in the Header section.

NOTE: Even if the *Fixed Header CRC* check fails and therefore the *Frame border count* value *b* is unknown from the Header section, the decoder may still extract the *Frame border count* from any *Frame border description* element carried within the Directory section.

5.3.1.2 xHE-AAC audio frame

An xHE-AAC audio frame contains the encoded data for a fixed duration of source audio content. The size of each audio frame may vary within the boundaries of the bit reservoir (see clause 5.3.1.3).

The xHE-AAC audio frame has the following structure:

- USAC access unit $u \times 8$ bits.
- Audio frame content CRC 16 bits.

USAC access unit: this multi-byte field of variable length contains the USAC access unit content ("UsacFrame()") as defined by the MPEG standard (see ISO/IEC 23003-3 [10]), including all required dynamic configuration elements such as the Dynamic SBR configuration. The USAC access unit encoder should be configured such that the first bit of a UsacFrame(), i.e. the **usacIndependencyFlag**, is set to 1 at regular intervals (e.g. at least once within every audio super frame that contains at least one audio frame border), in order to allow for short tune-in delay and quick frame loss recovery; the audio frame in which the usacIndependencyFlag is set to 1 should appear as early as possible within the audio super frame.

Audio frame content CRC: this 16-bit CRC value is calculated over the *USAC access unit* field of the audio frame, as defined in annex D.

5.3.1.3 Transport of xHE-AAC audio frames within the payload section

The USAC access unit encoder generates a continuous sequence of audio frames at a constant bit rate over the long term.

The individual length of each audio frame in the continuous sequence is variable but constrained by the bit reservoir mechanism in the audio encoder to allow for improved audio quality. The encoder's bit reservoir buffer level is signalled to the decoder, to reduce required input buffer size and the extra tune-in delay to a minimum.

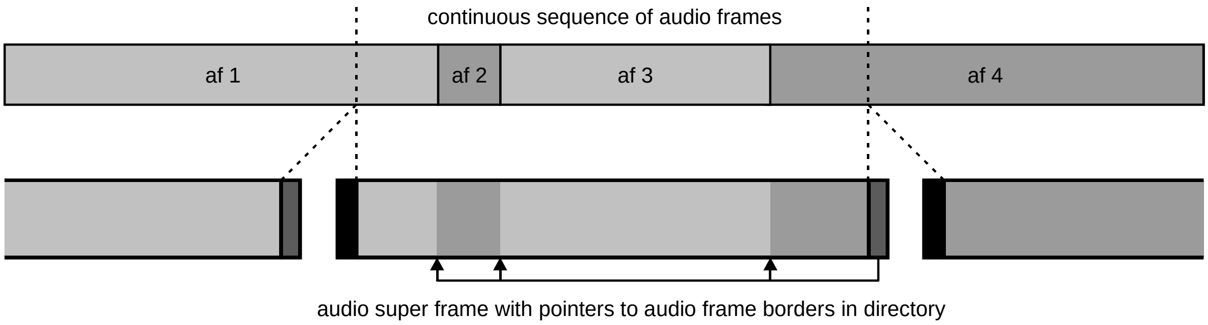


Figure 5: Example for splitting of audio frames into the audio super frame payload section

Audio frame transport

The xHE-AAC audio encoder generates a sequence of audio super frames (occupying one DRM logical frame for robustness modes A, B, C and D, or two DRM logical frames for robustness mode E). The audio frames as generated by the USAC access unit encoder are inserted into the Payload section of the audio super frame as a continuous byte sequence without any padding bytes in-between. Should padding be required to achieve the overall fixed bit rate and byte-alignment, it is inserted by the USAC access unit encoder into the audio frames themselves.

The frame borders of audio frames do not need to be and typically will not be aligned with the audio super frame boundaries. Instead, audio frames are not synchronized to audio super frames; they continue from the current audio super frame into the Payload section of subsequent audio super frame(s). The frame borders within the Payload section of an audio super frame can be derived from the Header and Directory section of the audio super frame (there may be none!).

An audio super frame shall not contain a *Frame border description* element without at least one byte of the corresponding audio frame data. If the available space in an audio super frame is not sufficient to hold at least 1 byte of the next audio frame in the Payload section plus the 2 bytes of the related extra *Frame border description* element in the Directory section, then the remaining space in the Payload section shall be filled with audio frame content, while the related *Frame border description* element is carried as the first *Frame border description* element in the Directory section of the following audio super frame (i.e. located at the end of the Directory section). The *Frame border index* value of such a delayed *Frame border description* element shall carry the special value 0xFFE or 0xFFF; with 0xFFF indicating the start of the audio frame at the last byte of the Payload section of the previous audio super frame. A decoder therefore always needs to buffer the last 2 bytes within the Payload section for a possible later processing along with the next audio super frame.

Bit reservoir mechanism

The bit reservoir mechanism is common to many MPEG audio codecs. It enables the generation of a fixed bit rate output stream over the long term while allowing the encoder to adjust the size of each audio frame to cater for the varying complexity of the input source signal.

The xHE-AAC audio encoder may - in addition to the average size of each audio frame derived from the stream bit rate and audio configuration - allocate additional bits from a bit reservoir to each audio frame (i.e. decrease the bit reservoir level, up to a minimum level of 0), or use less than the average number of bits for an audio frame, thereby increasing the bit reservoir level (up to the defined size of the bit reservoir buffer).

As a worst-case scenario, the xHE-AAC decoder can start decoding the received audio information as soon as the amount of data equal to the maximum possible audio frame size has been received. The size of the bit reservoir buffer (and thus the maximum possible audio frame size) is fixed to 6 144 bits × *number_of_channels*, with the *number_of_channels* being 1 or 2 depending on the *audio mode* parameter of the SDC data entity type 9 indicating mono or stereo, respectively.

To reduce and optimize the tune-in delay, the encoder's bit reservoir level is explicitly signalled to the xHE-AAC decoder by use of the *Bit reservoir level* field carried in the audio super frame's Header section. Based on this information the xHE-AAC decoder can typically start decoding much earlier as outlined below.

The bit reservoir level is transmitted as a quantized four-bit unsigned integer value. On the encoder side, the current *bitResLevel* value is first quantized and then rounded towards 0 for transmission:

$$Bit\ reservoir\ level = \left\lfloor \frac{bitResLevel}{384 \times number\ of\ channels} \right\rfloor$$

On the decoder side, the current *bitResLevel* value of the encoder can be reconstructed from the *Bit reservoir level* value carried in the Header section as follows:

$$bitResLevel = (Bit\ reservoir\ level + 1) \times\ 384\ \times number\ of\ channels$$

With this information the xHE-AAC decoder needs to buffer only the one audio frame to which *bitResLevel* refers plus *bitResLevel* bits from successive audio frames before decoding may start.

The fact that the bit reservoir fullness can only be transmitted in steps of 384 needs to be taken into account when designing the decoder input buffer. The size of the decoder input buffer shall therefore be at least 6 144 + 384 = 6 528 bits for mono signals, or twice this value for stereo signals (MPEG standard for AAC and USAC), independent from whether the MPEG Surround option is employed or not.

NOTE: This input buffer size is required to deal with the bit reservoir mechanism and to re-assemble audio frames; the additional decoder buffer required for handling received audio super frames is not considered here.

The tune-in delay caused by the bit reservoir mechanism varies according to the bit rate of the audio stream:

- Typically ≤ 0,8 seconds with a maximum of 1,2 seconds for an 8 kbps mono transmission in robustness modes A, B, C and D.
- Typically ≤ 0,4 seconds with a maximum of 0,6 seconds for a 32 kbps stereo transmission in robustness mode E.

EXAMPLE 1: An 8 kbps mono audio stream for robustness modes A, B, C and D requires 0,80 seconds of audio content for a maximum-size audio frame (to fill the 6 144 bits of the decoder input buffer), which may as a worst-case scenario span 3 consecutive audio super frames of 400 ms each, resulting in up to 1,2 seconds tune-in time (at an average net audio bitrate of 7 660 bps).

EXAMPLE 2: A 32 kbps stereo audio stream for robustness mode E requires 0,39 seconds of audio content for a maximum-size audio frame (to fill the 2 × 6 144 bits decoder input buffer), which may as a worst-case scenario span 3 consecutive audio super frames of 200 ms each, resulting in up to 0,6 seconds tune-in time (at an average net audio bitrate of 31 320 bps).

Dynamic reconfiguration

Pending reconfigurations of the audio stream configuration shall be pre-announced to the xHE-AAC audio encoder, just as they are announced to the receiver by the FAC *Reconfiguration index* field. It is the audio encoder's responsibility to finish the Payload section with the completed final audio frame when creating the last audio super frame using the old audio configuration. The decoder input buffer fill level should reach 0 before the point of reconfiguration and shall start with fill level 0 after the reconfiguration (to minimize the required tune-in delay caused by buffering at the audio decoder).

5.3.2 xHE-AAC decoder configuration

For xHE-AAC, the majority of audio configuration parameters shall automatically be determined by the xHE-AAC audio encoder based on its internal optimized and tuned parameter sets. This approach allows for maximum flexibility for future improvements of xHE-AAC encoder implementations, while reducing the broadcaster's burden to decide about optimized audio parameter sets. To free the audio frames from carrying repetitive static configuration information and thus grant a maximum of the available MSC bit rate to the actual encoded audio content, this static configuration information shall be transmitted in the SDC data entity type 9.

Parameters to be defined by the broadcaster are restricted to:

- MSC stream bit rate;
- the choice of mono or stereo audio (to be transmitted in the *audio mode* field); and
- the optional use of MPEG Surround for stereo-compatible surround sound (to be signalled in the *coder field*).

The xHE-AAC audio encoder shall determine upon start-up - based on the audio parameters configured by the broadcaster - an optimized set of static tools and configuration elements. This static configuration comprises:

- the sampling rate of the audio codec (to be transmitted in the *audio sampling rate* field); and
- the xHE-AAC Static Config (to be transmitted in the *codec specific config* field).

Only a few configuration elements - if present - are carried within the xHE-AAC audio frames (as described in ISO/IEC 23003-3 [10]), as their values might need to change dynamically between audio frames. An example of such an element is the SBR Dynamic Config.

The xHE-AAC Static Config is a set of full bytes of variable length carried from the xHE-AAC encoder to the xHE-AAC decoder in the SDC data entity type 9 *codec specific config* field. It roughly follows the UsacConfig (as defined in ISO/IEC 23003-3 [10]), but leaves out elements not required or statically defined for xHE-AAC in DRM.

The xHE-AAC Static Config ("xHEAACStaticConfig()") has the following syntax elements, data elements and syntax:

Syntax Elements

xHEAACDecoderConfig() This element contains all information required by the decoder to interpret the bit stream (along with the remaining parameters in SDC data entity type 9).

xHEAACSingleChannelElementConfig()
Contains all information needed for configuring the decoder to decode one single channel (i.e. mono transmission).

xHEAACChannelPairElementConfig()
In analogy to the above this element configuration contains all information needed for configuring the decoder to decode one channel pair (i.e. stereo transmission).

xHEAACMps212Config() All set-up parameters for the MPS212 parametric stereo coding tools are assembled in this syntax element.

UsacConfigExtensionDrm() Bitstream structure for adding extensions to the xHEAACStaticConfig(). Table 3A lists the allowed types of extensions.

Table 3A: Value of usacConfigExtTypeDrm

usacConfigExtTypeDrm	Value
ID_CONFIG_EXT_LOUDNESS_INFO_DRM	0
ID_CONFIG_EXT_LOUDNESS_INFO_DRM_SAMPLEPEAK	1
/* reserved for future use */	2 and higher

loudnessInfoSetDrm() Bitstream structure in analogy to the loudnessInfoSet() structure as defined in ISO/IEC 23003-4 [8].

Data Elements

Unless otherwise stated, all data elements printed in **bold** in the following syntax description and all syntax elements not defined in the present document shall follow the definition given in ISO/IEC 23003-3 [10].

Table 4: Syntax of xHEAACStaticConfig ()

Syntax	No. of bits	Note
xHEAACStaticConfig()		
{		
coreSbrFrameLengthIndexDrm;	2	Uimsbf, 1
xHEAACDecoderConfig();		
if (usacConfigExtensionPresent==1) {	1	
UsacConfigExtensionDrm();		
}		
}		
NOTE: "**coreSbrFrameLengthIndexDrm+1**" is equivalent to the value of **coreSbrFrameLengthIndex** in ISO/IEC 23003-3 [10] (i.e. **coreSbrFrameLengthIndexDrm**==0 is equivalent to **coreSbrFrameLengthIndex**==1).		

Table 5: Syntax of xHEAACDecoderConfig()

Syntax	No. of bits	Note
xHEAACDecoderConfig()		
{		
elemIdx = 0;		
switch (*audio mode*) {		1
case: "00"		
usacElementType**[**elemIdx**] =** ID_USAC_SCE;		
xHEAACSingleChannelElementConfig(sbrRatioIndex)		2
break;		
case: "10"		
usacElementType**[**elemIdx**] =** ID_USAC_CPE;		
xHEAACChannelPairElementConfig(sbrRatioIndex)		2
break;		
default:		
break;		
}		
numExtElements = escapedValue(2,4,8)**;**		
for (elemIdx=1; elemIdx<=numExtElements; ++elemIdx) {		
usacElementType**[**elemIdx**] =** ID_USAC_EXT		
UsacExtElementConfig();		
}		
}		
NOTE 1: xHEAACSingleChannelElementConfig(), xHEAACChannelPairElementConfig(),and UsacExtElementConfig() signalled at position elemIdx refer to the corresponding elements in UsacFrame() at the respective position elemIdx.		
NOTE 2: sbrRatioIndex can be derived from the **coreSbrFrameLengthIndexDrm** value.		

Table 6: Syntax of xHEAACSingleChannelElementConfig()

Syntax	No. of bits	Note
xHEAACSingleChannelElementConfig(sbrRatioIndex)		
{		
noiseFilling;	**1**	
if (sbrRatioIndex > 0) {		
SbrConfig();		
}		
}		

Table 7: Syntax of xHEAACChannelPairElementConfig()

Syntax	No. of bits	Note
xHEAACChannelPairElementConfig(sbrRatioIndex)		
{		
noiseFilling;	**1**	
if (sbrRatioIndex > 0) {		
SbrConfig();		
stereoConfigIndex;	**2**	**Uimsbf**
}		
else {		
stereoConfigIndex = 0;		
}		
if (stereoConfigIndex > 0) {		
xHEAACMps212Config(stereoConfigIndex);		
}		
}		

Table 8: Syntax of xHEAACMps212Config()

Syntax	No. of bits	Note
xHEAACMps212Config(stereoConfigIndex)		
{		
bsFreqRes;	**3**	**Uimsbf**
bsFixedGainDMX	**3**	**Uimsbf**
bsTempShapeConfigDrm;	**1**	**Uimsbf,** 1
bsHighRateMode;	**1**	**Uimsbf**
bsPhaseCoding;	**1**	**Uimsbf**
bsOttBandsPhasePresent;	**1**	**Uimsbf,** 2
if (bsOttBandsPhasePresent) {		
bsOttBandsPhase;	**5**	**Uimsbf**
}		
if (bsResidualCoding) {		3
bsResidualBands;	**5**	**Uimsbf**
bsOttBandsPhase = max(bsOttBandsPhase,bsResidualBands);		
bsPseudoLr;	**1**	**Uimsbf**
}		
}		

NOTE 1: **bsTempShapeConfigDrm**==1 is equivalent to **bsTempShapeConfig**==3 in ISO/IEC 23003-3 [10] (i.e. TSD tool enabled).
NOTE 2: if **bsOttBandsPhasePresent**==0, **bsOttBandsPhase** is initialized according to table 104 in ISO/IEC 23003-3 [10].
NOTE 3: **bsResidualCoding** depends on **stereoConfigIndex** according to table 72 in ISO/IEC 23003-3 [10].

Table 8A: Syntax of UsacConfigExtensionDrm()

Syntax	No. of bits	Mnemonic
UsacConfigExtensionDrm()		
{		
numConfigExtensions = escapedValue(2,4,8) + 1;		
for (confExtIdx=0; confExtIdx<numConfigExtensions; confExtIdx++) {		
usacConfigExtTypeDrm[confExtIdx] = escapedValue(3,5,8);		
usacConfigExtLengthDrm[confExtIdx] = escapedValue(3,5,8);		
switch (usacConfigExtTypeDrm[confExtIdx]) {		
case ID_CONFIG_EXT_LOUDNESS_INFO_DRM:		
loudnessInfoSetDrm(0);		
break;		
case ID_CONFIG_EXT_LOUDNESS_INFO_DRM_SAMPLEPEAK:		
loudnessInfoSetDrm(1);		
break;		
default:		
while (usacConfigExtLengthDrm[confExtIdx]--) {		
tmp;	**8**	**uimsbf**
}		
break;		
}		
}		
}		

Table 8B: Syntax of loudnessInfoSetDrm()

Syntax	No. of bits	Mnemonic
loudnessInfoSetDrm(samplePeakLevelPresent)		
{		
loudnessInfoAlbumCount=0;		
loudnessInfoCount=1;		
loudnessInfoSetExtPresent=0;		
/* loudnessInfo[0] */		
{		
drcSetId=0;		
downmixId=0;		
if (samplePeakLevelPresent==1) {		
samplePeakLevel = 20 – 0,5 × **bsSamplePeakLevel**; (note 1)	**8**	**uimsbf**
} else {		
samplePeakLevel = 0;		
}		
truePeakLevelPresent=0;		
measurementCount=1;		
/* measurement[0] */		
{		
methodDefinition=1; /* program loudness */		
methodValue = –57,75 + 0,25 × **bsMethodValue**; (note 2)	**8**	**uimsbf**
measurementSystem=2; /* ITU-R BS.1770-4 */		
reliability=3; /* measured and accurate */		
}		
}		
}		
NOTE 1: samplePeakLevel can take values in the range from -107,5 dBFS to 20 dBFS, with 0,5 step size.		
NOTE 2: methodValue can take values in the range from -57,75 LKFS to 6 LKFS, with 0,25 step size.		

Table 8C: Syntax of escapedValue()

Syntax	No. of bits	Mnemonic
escapedValue(nBits1, nBits2, nBits3)		
{		
value;	**nBits1**	**uimsbf**
if (value == 2^{nBits1}-1) {		
value += **valueAdd;**	**nBits2**	**uimsbf**
if (valueAdd == 2^{nBits2}-1) {		
value += **valueAdd;**	**nBits3**	**uimsbf**
}		
}		
return value;		
}		

5.3.3 xHE-AAC error concealment

5.3.3.0 Introduction

The xHE-AAC decoder shall include concealment functionality for at least one missing or corrupt audio frame. The recommended method for concealment is interpolation between two valid audio frames, which will increase the delay of the decoder by one audio frame.

There are various tests inside the core decoder, starting with the CRC test and ending in a variety of plausibility checks. If such a check indicates an invalid audio frame, then concealment is applied. Concealment is also applied when the channel decoder indicates a distorted data frame.

xHE-AAC consists of several codec modules, to which individual concealment methods may be applied. The following clauses give an overview of the key modules and the applicable concealment methods, supplemented by xHE-AAC specifics.

5.3.3.1 Frequency Domain coding (AAC based coding and TCX)

Concealment works on the spectral data just before the final frequency to time conversion. If the optional delay of one frame is used, concealment may interpolate between the preceding and the following valid frames to create the data for the missing frame, in case a single frame is corrupted. If multiple frames are corrupted (e.g. more than one or two consecutive frames), concealment implements a fade out. If the decoder recovers from the error condition, the concealment algorithm performs a fade-in. Fade-in might be slightly delayed (suppressed) to deal with error conditions where only a valid frame here and there is received.

For TCX in the linear prediction processing path of xHE-AAC also the techniques described in AMR-WB+ [11] may be used.

Interpolation of one corrupt frame (optional)

In the following, the current audio frame is frame number *n*, the corrupt audio frame to be interpolated is the frame *n*-1 and the frame before has the number *n*-2. Frame number *n*-2 is the preceding valid frame which spectral values have been stored during the processing in the previous call to the decoder.

The determination of window sequence and the window shape of the corrupt frame are described in table 9.

Table 9: Interpolated window sequences and window shapes

window sequence *n-2*	window sequence *n*	window sequence *n*-1	window shape *n*-1
ONLY_LONG_SEQUENCE or LONG_START_SEQUENCE or LONG_STOP_SEQUENCE or STOP_START_SEQUENCE	ONLY_LONG_SEQUENCE or LONG_START_SEQUENCE or LONG_STOP_SEQUENCE or STOP_START_SEQUENCE	ONLY_LONG_SEQUENCE	0
ONLY_LONG_SEQUENCE or LONG_START_SEQUENCE or LONG_STOP_SEQUENCE or STOP_START_SEQUENCE	EIGHT_SHORT_SEQUENCE	LONG_START_SEQUENCE	1
EIGHT_SHORT_SEQUENCE	EIGHT_SHORT_SEQUENCE	EIGHT_SHORT_SEQUENCE	1
EIGHT_SHORT SEQUENCE	ONLY_LONG_SEQUENCE or LONG_START_SEQUENCE or LONG_STOP_SEQUENCE or STOP_START_SEQUENCE	LONG_STOP_SEQUENCE	0
NOTE: The window sequences are depicted in ISO/IEC 23003-3 [10], table 88 - Window Sequences and Transform windows dependent of coreCoderFrameLength (ccfl).			

The scalefactor band energies of frames *n*-2 and *n* are calculated. If the window sequence in one of these frames is an EIGHT_SHORT_SEQUENCE and the final window sequence for frame *n*-1 is one of the long transform windows, the scalefactor band energies are calculated for long block scalefactor bands by mapping the frequency line index of short block spectral coefficients to a long block representation. The new interpolated spectrum is built on a per-scalefactorband basis by reusing the spectrum of the older frame *n*-2 and multiplying a factor to each spectral coefficient. An exception is made in the case of a short window sequence in frame *n*-2 and a long window sequence in frame *n*, here the spectrum of the actual frame n is modified by the interpolation factor. This factor is constant over the range of each scalefactor band and is derived from the scalefactor band energy differences of frames *n*-2 and *n*. Finally noise substitution is applied by flipping the sign of the interpolated spectral coefficients randomly.

Fade-out and fade-in

Fade-out and fade-in behaviour, i.e. the attenuation ramp, might be fixed or adjustable by the user. The spectral coefficients from the last frame are attenuated by a factor corresponding to the fade-out characteristics and then passed to the frequency to-time mapping. Depending on the attenuation ramp, the concealment switches to muting after a number of consecutive invalid frames, which means the complete spectrum will be set to 0.

After recovering from the error condition, the decoder fades in again depending on a ramp-up function possibly different from the ramp-down characteristics. If the concealment has switched to muting, fade-in might be suppressed for a configurable number of frames to avoid annoying output of non-consecutive single valid frames.

5.3.3.2 ACELP

Concealment methods described in AMR-WB+ [11] should be applied.

5.3.3.3 SBR

The SBR error concealment algorithm is based on using previous envelope and noise-floor values with an applied decay, as a substitute for the corrupt data. This also applies for the case where predictive vector coding is used.

In the flowchart of figure 6 the basic operation of the SBR error concealment algorithm is outlined. If the frame error flag is set, error concealment bitstream data is generated to be used instead of the corrupt bitstream data. The concealment data is generated according to the following.

The time frequency grids are set to:

$$L_E = 1$$

$$\mathbf{t}_E(0) = \mathbf{t}'_E(L'_E) - numTimeSlots$$

$$\mathbf{t}_E(1) = numTimeSlots$$

$$\mathbf{r}(l) = HI \quad ,0 \le l < L_E$$

$$bs\_pointer = 0$$

$$L_Q = 1$$

$$\mathbf{t}_Q = \left[\mathbf{t}_E(0), \mathbf{t}_E(1)\right]$$

The delta coding direction for both the envelope data and noise-floor data are set to be in the time-direction. The envelope data is calculated according to:

$$\mathbf{E}_{Delta}(k,l) = \begin{cases} -step & ,\mathbf{E}_{prev}(k,l) > target \\ step & ,otherwise \end{cases} \quad ,0 \le k < \mathbf{n}(\mathbf{r}(l)), 0 \le l < L_E$$

where:

$$step = \begin{cases} 2 & ,if \quad bs\_amp\_res = 1 \\ 1 & ,otherwise \end{cases}$$

$$target = \begin{cases} \mathbf{panOffset}(bs\_amp\_res) & ,if \quad bs\_coupling = 1 \\ 0 & ,otherwise \end{cases}$$

And where *bs_amp_res* and *bs_coupling* are set to the values of the previous frame.

The noise floor data is calculated according to:

$$\mathbf{Q}_{Delta}(k,l) = 0 \quad ,\begin{cases} 0 \le l < L_Q \\ 0 \le k < N_Q \end{cases}$$

Furthermore, the inverse-filtering levels in *bs_invf_mode* are set to the values of the previous frame, and all elements in *bs_add_harmonic* are set to zero.

If the frame error is not set, the present time grid and envelope data may need modification if the previous frame was corrupt. If the previous frame was corrupt the time grid of the present frame is modified in order to make sure that there is a continuous transition between the frames. The envelope data for the first envelope is modified according to:

$$\mathbf{E}_{mod}\left(k,0\right)=\mathbf{E}\left(k,0\right)+a\cdot\log_2\left(\frac{\mathbf{t}_E\left(1\right)-\mathbf{t}_E\left(0\right)}{\mathbf{t}_E\left(1\right)-estimated\_start\_pos}\right),\quad 0\le k<\mathbf{F}\left(\mathbf{r}\left(l\right),0\right)$$

where:

$$estimated\_start\_pos=\mathbf{t}'_E\left(L'_E\right)-numberTimeSlots$$

After the delta coded data has been decoded, a plausibility check is performed to make sure that the decoded data is within reasonable limits. The required limits are:

For the envelope data the logarithmic values shall fulfil:

$$\mathbf{E}\left(k,l\right)\le\begin{cases}35 & ,ampRes=0\\ 70 & ,ampRes=1\end{cases}$$

otherwise the frame will be considered corrupt.

The time grids are also verified according to the following rules (if any of the below is true, the frame is considered to be corrupt):

$$L_E<1$$

$$L_E>8$$

$$L_Q>2$$

$$\mathbf{t}_E\left(0\right)<0$$

$$\mathbf{t}_E\left(0\right)\ge\mathbf{t}_E\left(L_E\right)$$

$$\mathbf{t}_E\left(0\right)>3$$

$$\mathbf{t}_E(L_E)<16$$

$$\mathbf{t}_E(L_E)>19$$

$$\mathbf{t}_E\left(l\right)\ge\mathbf{t}_E\left(l+1\right),0\le l<L_E$$

$$l_A>L_E$$

$$L_E=1\ AND\ L_Q>1$$

$$\mathbf{t}_Q\left(0\right)\ne\mathbf{t}_E\left(0\right)$$

$$\mathbf{t}_Q\left(L_Q\right)\ne\mathbf{t}_E\left(L_E\right)$$

$$\mathbf{t}_Q\left(l\right)\ge\mathbf{t}_Q\left(l+1\right),0\le l<L_Q$$

All elements of $\mathbf{t}_Q$ are not among the elements of $\mathbf{t}_{E.}$

If the plausibility check fails, the frame error flag is set and the error concealment outlined above is applied.

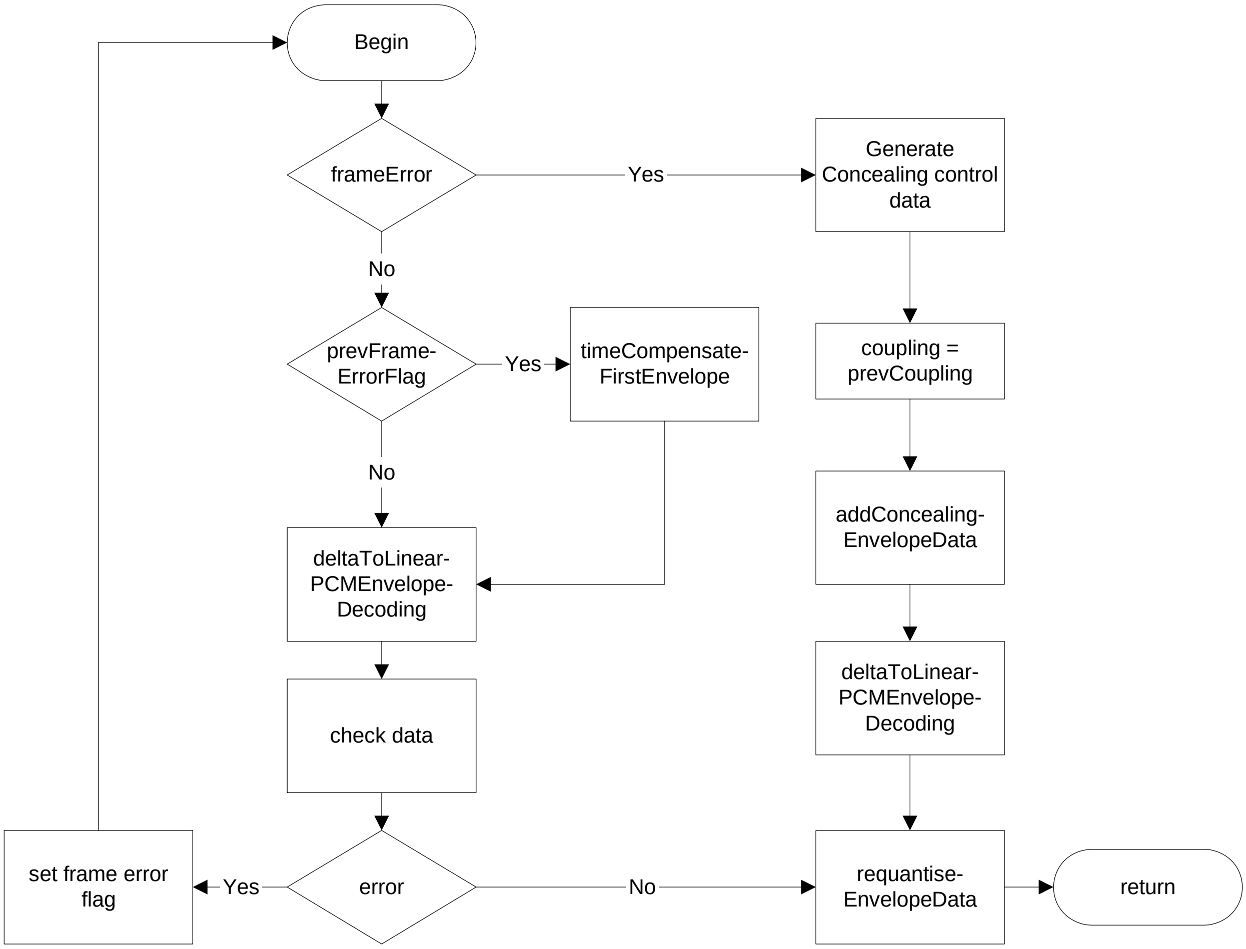


Figure 6: SBR error concealment overview

inter-TES (Temporal Envelope Shaping)

bs_temp_shape and bs_inter_temp_shape_mode are assumed to be zero in concealed frames.

5.3.3.4 MPS212 parametric stereo

MPS212 parametric stereo concealment is based on the fact that the stereo image is quasi-stationary. The concealment strategy keeps the MPS212 parametric stereo settings from the last valid frame until a new set of MPS212 parametric stereo settings can be decoded from a valid frame.

5.3.3.5 MDCT based Complex Prediction

Similar to MPS212 parametric stereo (see clause 5.3.3.4), the Complex Prediction concealment is based on the fact that the stereo image is quasi-stationary. The concealment strategy keeps the Complex Prediction settings from the last valid frame until a new set of Complex Prediction settings can be decoded from a valid frame.

5.3.3.6 Forward Aliasing Cancellation

The Forward Aliasing Cancellation signal part is assumed to be zero in concealed frames. During recovery the Forward Aliasing Cancellation data (fac_data()) is ignored if present.

5.3.4 xHE-AAC + MPS

The standard UsacExtElement() is used to carry MPEG Surround as defined by ISO/IEC 23003-3 [10]. The corresponding static part of the configuration is signalled via SDC data entity type 9 in the *codec specific configuration* field.

5.3.5 Loudness metadata

The UsacConfigExtensionDrm() is used to carry loudness metadata. The corresponding static part of the configuration is signalled via SDC data entity type 9 in the codec specific configuration field.

It is recommended that the loudness metadata be appropriately considered during the decoding process. Provisions applicable to decoding and application of loudness metadata as conveyed through the data element loudnessInfoSetDrm() are specified in ISO/IEC 23003-4 [8].

5.4 AAC coding

5.4.1 AAC

ISO/IEC 14496-3 [1] defines the MPEG-4 Audio standard. The audio coding standard MPEG-4 AAC is part of the MPEG-4 Audio standard. From the possible audio object types, only the Error Robust (ER) AAC Scalable object type (Object Type ID = 20), which is part of the High Quality Audio Profile, is used in the DRM system.

DRM specific usage of MPEG-4 AAC: Three error robustness tools may be used within an MPEG-4 ER AAC bitstream: HCR (Huffman Codeword Reordering), VCB11 (Virtual Codebooks for Codebook 11) and RVLC (Reversible Variable Length Coding). In the DRM system, all AAC bitstreams shall use the HCR tool, since this tool reduces the error sensitivity of the bitstream significantly with a minimum of overhead. The VCB11 tool shall be used, since for low bit rates, the VCB11 overhead is less than 1 %. The RVLC tool is not used, since it introduces a significant bit rate overhead that is a major drawback for the low bit rates used by DRM.

The MPEG-4 AAC tool PNS (Perceptual Noise Substitution) is not used in DRM since SBR provides this functionality more appropriately.

For DRM the 960 transform shall be used.

Robustness modes A, B, C and D:

- When 12 kHz sampling is used, 5 AAC frames shall be combined into one audio super frame.
- When 24 kHz sampling is used, 10 AAC frames shall be combined into one audio super frame.
- The AAC sampling rate shall be 24 kHz when the stereo mode is used.

Robustness mode E:

- When 24 kHz sampling is used, 5 AAC frames shall be combined into one audio super frame.
- When 48 kHz sampling is used, 10 AAC frames shall be combined into one audio super frame.

No standard extension_payload() shall be used and the only allowed extensions are SBR (signalled via SDC) and MPS (signalled via SDC).

The left and the right channel in one stereo audio frame are transmitted in an interleaved way to achieve a decreasing error sensitivity within the stereo frame.

Any DRM AAC bitstream can easily be translated into an MPEG-4 ER compliant bitstream by applying the above rules.

When the transmission is a base layer (the Base/Enhancement flag in the FAC is 0, see clause 6.3.3), the AAC frame corresponds to aac_scalable_main_element() as defined in the MPEG-4 standard ISO/IEC 14496-3 [1].

The MPEG-4 standard defines how the bits for one raw error robust AAC audio frame are stored. Each element of the error robust AAC bitstream is assigned an error sensitivity category. In the DRM system there are two possible error robust AAC audio frames.

Mono audio frame

One mono audio frame consists of three consecutive parts, hereinafter called mono1, mono2 and mono3. Mono1 contains the Side Information (SI) bits, mono2 contains the Temporal Noise Shaping (TNS) bits and mono3 contains the spectral data bits. The error sensitivity decreases from mono1 to mono3.

Stereo audio frame

One stereo audio frame consists of seven consecutive parts, hereinafter called stereo1 (common side info), stereo2 (side info left channel), stereo3 (side info right channel), stereo4 (TNS left channel), stereo5 (TNS right channel), stereo6 (spectral data left channel), stereo7 (spectral data right channel). With this interleaving of left and right channel, the error sensitivity is decreasing from stereo1 to stereo7.

Table 10: Syntax of aac_super_frame()

Syntax	No. of bits	Note
aac_super_frame(audio_info, robustness_mode) //audio info from the SDC		
{		
if (robustness_mode == A ¦ B ¦ C ¦ D) {		
switch (audio_info.audio_sampling_rate) { //only 12 000 and 24 000 is allowed		
case 12 000: num_frames = 5;		
break;		
case 24 000: num_frames = 10;		
break;		
}		
}		
else { //robustness_mode == E		
switch (audio_info.audio_sampling_rate) { //only 24 000 and 48 000 is allowed		
case 24 000: num_frames = 5;		
break;		
case 48 000: num_frames = 10;		
break;		
}		
}		
aac_super_frame_header(num_frames - 1);		
for (f = 0; f < num_frames; f++) {		
aac_crc_bits[f]	8	see annex D
}		
for (f = 0; f < num_frames; f++) {		
for (b = 0; b < num_bytes; b++)		
audio_frame[f][b]	8	
}		
}		
NOTE: audio_frame is either an AAC or an AAC + SBR frame including possible enhancements.		

Table 11: Syntax of aac_super_frame_header()

Syntax	No. of bits	Note
aac_super_frame_header(num_borders)		
{		
previous_border = 0;		
for (n = 0; n < num_borders; n++) {		
frame_length[n] = **frame_border** - previous_border; // frame border in bytes	12	2
previous_border = frame_border;		
}		
frame_length[num_borders] = audio_payload_length - previous_border;		
if (num_borders == 9)		
reserved // byte-alignment	4	
}		
NOTE 1: The audio_payload_length is derived from the length of the audio super frame (data_length_of_part_A + data_length_of_part_B) subtracting the audio super frame overhead (bytes used for the audio super frame header() and for the aac_crc_bits).		
NOTE 2: If a value of a frame_border exceeds 4 095 bytes, only the 12 least significant bits are signalled. This condition is detected in the receiver implicitly because frame borders always increase from frame to frame.		

Header

The header contains information to recover the frame lengths of the num_frames AAC frames stored in the audio super frame.

All the frame lengths are derived from the absolute positions of the frame borders. These frame borders are stored consecutively in the header. Each frame border occupies 12 bits (unsigned integer, most significant bit first). The frame border is measured in bytes from the start of the AAC bitstream sequence. 4 padding bits are added in case num_frames==10. num_frames-1 frame borders are stored in the header.

CRC block

8-bit CRC check words derived from the CRC-bits of each corresponding AAC frame follow (see annex D for CRC calculation). For a mono signal, the CRC-bits cover (mono1, mono2). For a stereo signal, the CRC-bits cover (stereo1, stereo2, stereo3, stereo4, stereo5).

Payload part

The bytes of the AAC frames are stored consecutively.

Figure 7 illustrates an example audio super frame with 10 audio frames.

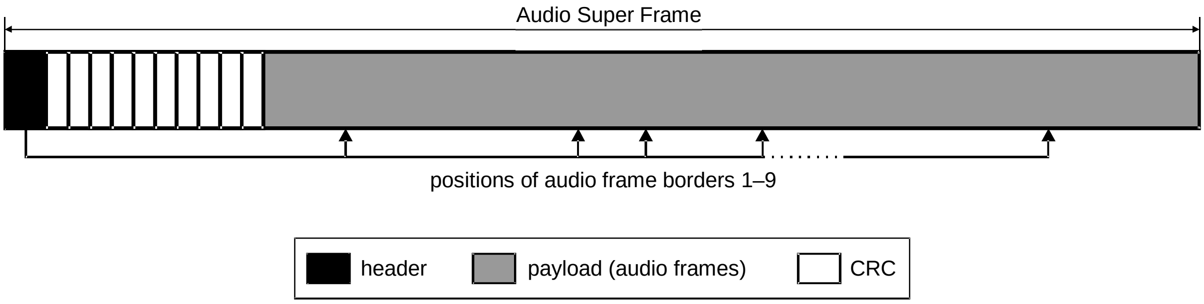


Figure 7: Example AAC audio super frame with 10 audio frames

5.4.2 AAC + SBR

The SBR sampling rate is twice the AAC sampling rate. One raw AAC + SBR frame contains an AAC part and a SBR part. The SBR part of the data is located at the end of the frame. The first bit in the SBR-bitstream is the last bit in the frame, and the SBR bits are thus written/read in reverse order. In this way, the starting points of respective part of the frame data are always easily found.

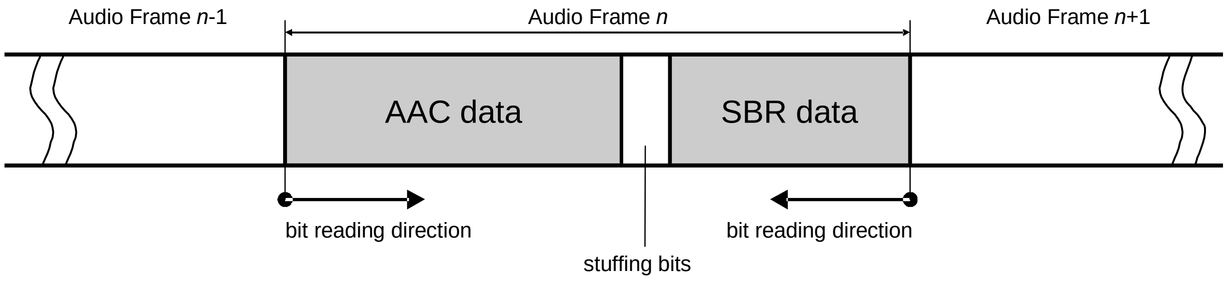


Figure 8: AAC + SBR frame

Both AAC and SBR data-sizes vary from frame to frame. The total size of the individual frames, now including the SBR data, can be derived from the aac_super_frame_header() as described in clause 5.3.1. Thus no extra signalling due to the varying SBR bit rate is needed.

The AAC + SBR frames are inserted into the audio super frame in the same manner as when SBR is not used.

The SBR tool as it is defined in the MPEG-4 Audio standard [1] is used. The method to extract the SBR elements from the SBR data and thus the mapping to the format defined in the MPEG-4 standard is given in table 12.

Table 12: Syntax of sbr_aac_frame()

Syntax	No. of bits	Note

sbr_aac_frame(audio_mode) // audio_mode is located in the SDC
{
sbr_crc_bits
 	
if (audio_mode != stereo)
sbr_extension_data(ID_SCE, 0);
else
sbr_extension_data(ID_CPE, 0);
}
	
	8
		
		see annex D

NOTE 1: sbr_extension_data() is defined in the MPEG-4 Audio standard [1]. NOTE 2: sbr_extension_data() uses a variable cnt for the num_align_bits calculation. cnt is not available in the DRM bitstream format and num_align_bits is 0 if bs_extended_data=0 or num_align_bits is the value of num_bits_left after leaving the while (num_bits_left > 7) loop if bs_extended_data=1.		

sbr_crc_bits Cyclic redundancy checksum for the SBR bit stream part. The CRC algorithm is applied to all the sbr_extension_data_bits().

5.4.3 Parametric Stereo coding

For improved performance at low bit rate stereo coding, a Parametric Stereo (PS) coder partly based upon the SBR framework is available. The MPEG PS tool as specified in MPEG-4 Audio [1] is used.

The general idea with PS coding is to send data describing the stereo image as side information along with a downmixed mono signal. This stereo side information is very concise and only requires a small fraction of the total bit rate allowing the mono signal to have maximum quality for the total bit rate given.

The PS coding method here defined, combines frequency selective panorama technique with a stereo ambience reconstruction technique. This allows a stereo image reconstruction well suited for both loudspeaker and headphones playback.

The stereo synthesis at the decoder reconstructs spatial properties but does not affect the total spectral energy. Hence, there is no colorization of the frequency spectrum compared to the mono compatible core signal. Further advantage is the backward compatibility which allows a decoder not supporting PS to successfully decode the mono core. This is possible as the PS data is included in the extended data field in the bitstream which optionally can be ignored.

The design target bit rates for applying parametric stereo coding to AAC + SBR are in the range 18 kbit/s to 26 kbit/s, however the technique may be used at any bit rate.

The Parametric Stereo (PS) data is conveyed in the SBR extended data field. PS signals a unique ID in the SBR extended data field defined by the bitstream element **bs_extension_id**. To be successfully decoded, PS needs to receive data from single channel elements in the bitstream, i.e. from a mono bitstream.

The function sbr_extension() used in sbr_channel_pair_base_element() and sbr_channel_pair_element() described in MPEG-4 Audio, clause "Payloads for the audio object type SBR", is defined as follows.

Table 13: Syntax of sbr_extension()

Syntax	No. of bits	Note

sbr_extension(bs_extension_id, num_bits_left)
{
    switch(bs_extension_id)
    {
        case MPEG_PARAMETRIC_STEREO:
            num_bits_left -= ps_data();
            break;
        default:
            bs_fill_bits;
            num_bits_left = 0;
            break;
    }
}
	
	num_bits_left
		
		1, 2
		
		bslbf, 3
		1

NOTE 1: The variable *num_bits_left* is the same as used in the sbr_pair_base_element() and sbr_channel_pair_element(). NOTE 2: ps_data() is defined in table 8.9 in MPEG-4 [1] and returns the total number of bits read. NOTE 3: bs_extension_id is defined in table 14.		

bs_extension_id Holds an ID of the extended data element.

Table 14: Definition of bs_extension_id

bs_extension_id	Meaning	Note
0	Shall not be used	
1	Reserved ID for future use	
2	MPEG PARAMETRIC STEREO	
3	Reserved ID for future use	

5.4.4 AAC error concealment

5.4.4.0 Introduction

The AAC core decoder includes a concealment function that increases the delay of the decoder by one frame.

There are various tests inside the core decoder, starting with the CRC test and ending in a variety of plausibility checks. If such a check indicates an invalid bit stream, then concealment is applied. Concealment is also applied when the channel decoder indicates a distorted data frame.

Concealment works on the spectral data just before the final frequency to time conversion. In case a single frame is corrupted, concealment interpolates between the preceding and the following valid frames to create the spectral data for the missing frame. If multiple frames are corrupted, concealment implements first a fade out based on slightly modified spectral values from the last valid frame. If the decoder recovers from the error condition, the concealment algorithm performs a fade-in on valid spectral values. Fade in might be delayed (suppressed) to deal with error conditions, where only a valid frame here and there is perceived.

5.4.4.1 Interpolation of one corrupt frame

In the following, the current frame is frame number n, the corrupt frame to be interpolated is the frame n-1 and the frame before has the number n-2. Frame number n-2 is the preceding valid frame which spectral values have been stored during the processing in the previous call to the decoder.

The determination of window sequence and the window shape of the corrupt frame is described in table 15.

Table 15: Interpolated window sequences and window shapes

window sequence *n-2*	window sequence *n*	window sequence *n*-1	window shape *n*-1
ONLY_LONG_SEQUENCE or LONG_START_SEQUENCE or LONG_STOP_SEQUENCE	ONLY_LONG_SEQUENCE or LONG_START_SEQUENCE or LONG_STOP_SEQUENCE	ONLY_LONG_SEQUENCE	0
ONLY_LONG_SEQUENCE or LONG_START_SEQUENCE or LONG_STOP_SEQUENCE	EIGHT_SHORT_SEQUENCE	LONG_START_SEQUENCE	1
EIGHT_SHORT_SEQUENCE	EIGHT_SHORT_SEQUENCE	EIGHT_SHORT_SEQUENCE	1
EIGHT_SHORT SEQUENCE	ONLY_LONG_SEQUENCE or LONG_START_SEQUENCE or LONG_STOP_SEQUENCE	LONG_STOP_SEQUENCE	0
NOTE: The window sequences are depicted in ISO/IEC 14496-3 [1], table 4.109 - Window Sequences.			

The scalefactor band energies of frames *n*-2 and *n* are calculated. If the window sequence in one of these frames is an EIGHT_SHORT_SEQUENCE and the final window sequence for frame *n*-1 is one of the long transform windows, the scalefactor band energies are calculated for long block scalefactor bands by mapping the frequency line index of short block spectral coefficients to a long block representation. The new interpolated spectrum is built on a per-scalefactorband basis by reusing the spectrum of the older frame *n*-2 and multiplying a factor to each spectral coefficient. An exception is made in the case of a short window sequence in frame *n*-2 and a long window sequence in frame *n*, here the spectrum of the actual frame n is modified by the interpolation factor. This factor is constant over the range of each scalefactor band and is derived from the scalefactor band energy differences of frames *n*-2 and *n*. Finally noise substitution is applied by flipping the sign of the interpolated spectral coefficients randomly.

5.4.4.2 Fade-out and fade-in

Fade-out and fade-in behaviour, i.e. the attenuation ramp, might be fixed or adjustable by the user. The spectral coefficients from the last frame are attenuated by a factor corresponding to the fade-out characteristics and then passed to the frequency to-time mapping. Depending on the attenuation ramp, the concealment switches to muting after a number of consecutive invalid frames, which means the complete spectrum will be set to 0.

After recovering from the error condition, the decoder fades in again depending on a ramp-up function possibly different from the ramp-down characteristics. If the concealment has switched to muting, fade-in might be suppressed for a configurable number of frames to avoid annoying output of non-consecutive single valid frames.

5.4.4.3 Concealment granularity

In case the spectral data is only partly destroyed, i.e. the CRC test and the plausibility checks are OK, error concealment might be applied in a finer granularity. The use of the error robustness tools HCR and VCB11 allow the decoder to detect invalid spectral lines. In case only a few lines are destroyed, the AAC concealment strategy above might be applied only to the corresponding scalefactor bands or to the destroyed lines.

5.4.4.4 SBR error concealment

The SBR error concealment algorithm is based on using previous envelope and noise-floor values with an applied decay, as a substitute for the corrupt data. In the flowchart of figure 9 the basic operation of the SBR error concealment algorithm is outlined. If the frame error flag is set, error concealment bitstream data is generated to be used instead of the corrupt bitstream data. The concealment data is generated according to the following.

The time frequency grids are set to:

$$L_E = 1$$

$$\mathbf{t}_E(0) = \mathbf{t'}_E(L'_E) - numTimeSlots$$

$$\mathbf{t}_E(1) = numTimeSlots$$

$$\mathbf{r}(l) = HI \quad ,0 \le l < L_E$$

$$bs\_pointer = 0$$

$$L_Q = 1$$

$$\mathbf{t}_Q = \left[\mathbf{t}_E(0), \mathbf{t}_E(1)\right]$$

The delta coding direction for both the envelope data and noise-floor data are set to be in the time-direction. The envelope data is calculated according to:

$$\mathbf{E}_{Delta}(k,l) = \begin{cases} -step & ,\mathbf{E}_{prev}(k,l) > target \\ step & ,otherwise \end{cases} \quad ,0 \le k < \mathbf{n}(\mathbf{r}(l)), 0 \le l < L_E$$

where:

$$step = \begin{cases} 2 & ,if \quad bs\_amp\_res = 1 \\ 1 & ,otherwise \end{cases}$$

$$target = \begin{cases} \mathbf{panOffset}(bs\_amp\_res) & ,if \quad bs\_coupling = 1 \\ 0 & ,otherwise \end{cases}$$

And where *bs_amp_res* and *bs_coupling* are set to the values of the previous frame.

The noise floor data is calculated according to:

$$\mathbf{Q}_{Delta}(k,l) = 0 \quad , \begin{cases} 0 \le l < L_Q \\ 0 \le k < N_Q \end{cases}$$

Furthermore, the inverse-filtering levels in *bs_invf_mode* are set to the values of the previous frame, and all elements in *bs_add_harmonic* are set to zero.

If the frame error is not set, the present time grid and envelope data may need modification if the previous frame was corrupt. If the previous frame was corrupt the time grid of the present frame is modified in order to make sure that there is a continuous transition between the frames. The envelope data for the first envelope is modified according to:

$$\mathbf{E}_{mod}(k,0) = \mathbf{E}(k,0) + a \cdot \log_2\left(\frac{\mathbf{t}_E(1) - \mathbf{t}_E(0)}{\mathbf{t}_E(1) - estimated\_start\_pos}\right), \quad 0 \le k < \mathbf{F}(\mathbf{r}(l), 0)$$

where:

$$estimated\_start\_pos = \mathbf{t}'_E(L'_E) - numberTimeSlots \; .$$

After the delta coded data has been decoded, a plausibility check is performed to make sure that the decoded data is within reasonable limits. The required limits are:

- for the envelope data the logarithmic values shall fulfil:

$$\mathbf{E}(k,l) \le \begin{cases} 35 & ,ampRes = 0 \\ 70 & ,ampRes = 1 \end{cases}$$

otherwise the frame will be considered corrupt.

The time grids are also verified according to the following rules (if any of the below is true the frame is considered to be corrupt):

$$L_E < 1$$

$$L_E > 5$$

$$L_Q > 2$$

$$\mathbf{t}_E(0) < 0$$

$$\mathbf{t}_E(0) \geq \mathbf{t}_E(L_E)$$

$$\mathbf{t}_E(0) > 3$$

$$\mathbf{t}_E(L_E) < 16$$

$$\mathbf{t}_E(L_E) > 19$$

$$\mathbf{t}_E(l) \geq \mathbf{t}_E(l+1), 0 \leq l < L_E$$

$$l_A > L_E$$

$$L_E = 1 \; AND \; L_Q > 1$$

$$\mathbf{t}_Q(0) \neq \mathbf{t}_E(0)$$

$$\mathbf{t}_Q(L_Q) \neq \mathbf{t}_E(L_E)$$

$$\mathbf{t}_Q(l) \geq \mathbf{t}_Q(l+1), 0 \leq l < L_Q$$

All elements of $\mathbf{t}_Q$ are not among the elements of $\mathbf{t}_{E.}$

If the plausibility check fails, the frame error flag is set and the error concealment outlined above is applied.

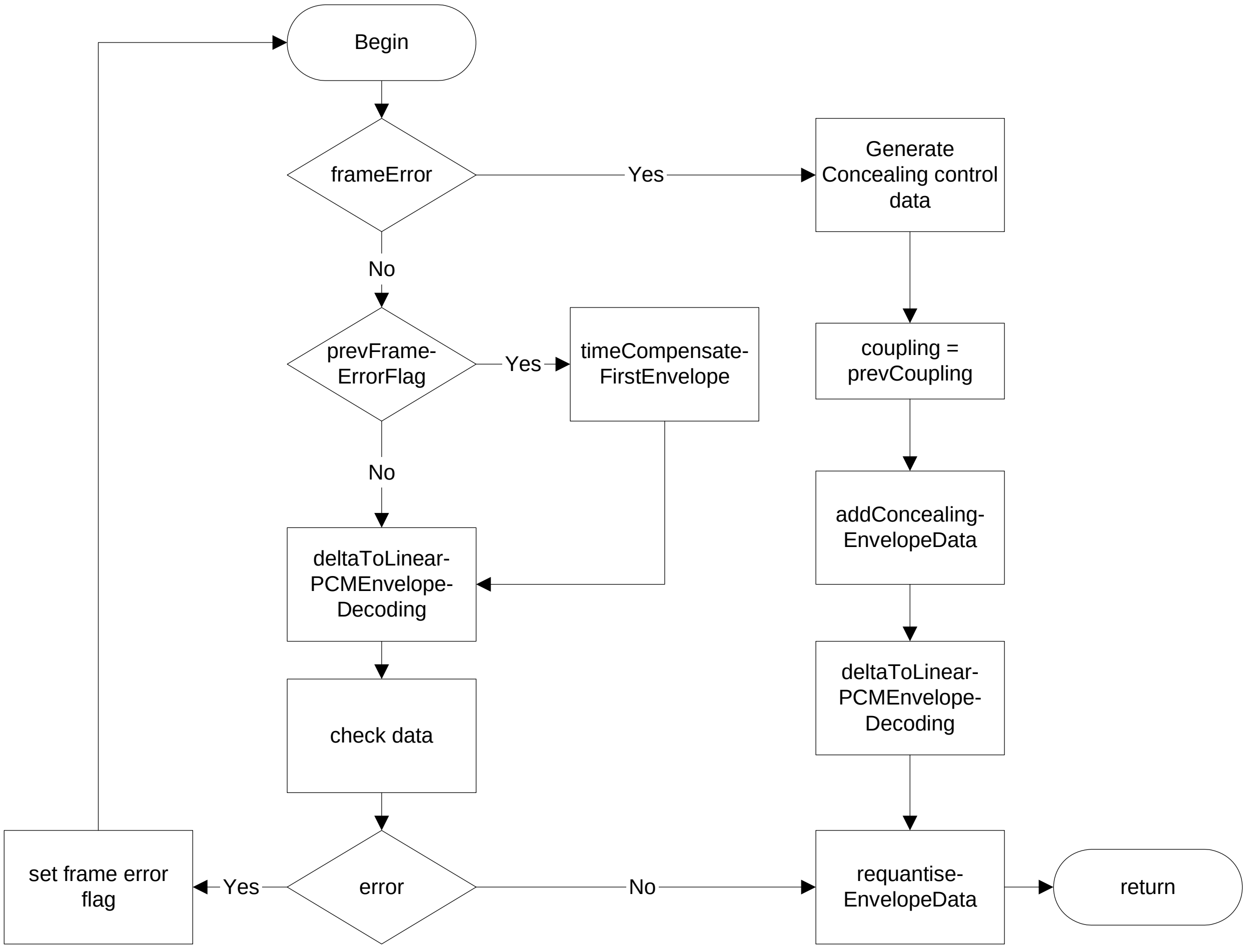


Figure 9: SBR error concealment overview

5.4.4.5 Parametric Stereo concealment

Parametric stereo concealment is based on the fact that the stereo image is quasi-stationary. The concealment strategy keeps the Parametric Stereo settings from the last valid frame until a new set of Parametric Stereo settings can be decoded from a valid frame.

5.4.5 AAC + MPS

AAC frames and AAC + SBR frames can be enhanced with MPS data. The MPS part of the audio frame follows immediately the AAC part and the presence of MPS data is signalled in the SDC data entity 9. If the MPEG Surround mode is not 000 then MPS data is available in the audio frame. The MPS data bits are written/read in the same order as the AAC data bits.

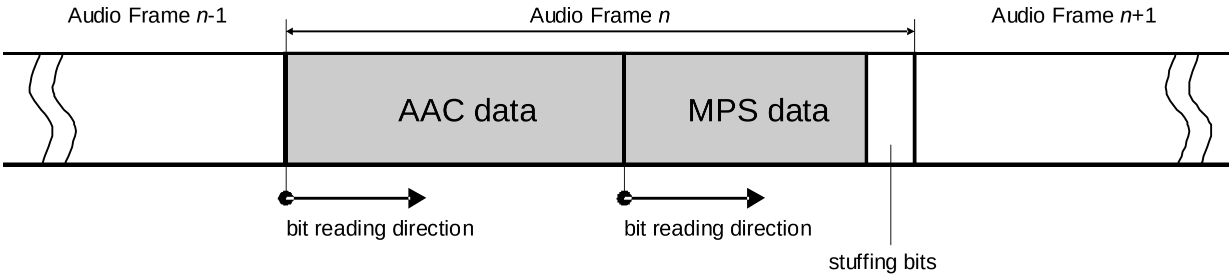


Figure 10: AAC + MPS frame

If both MPS and SBR data is present in an audio frame, stuffing bits are inserted between the MPS and SBR data.

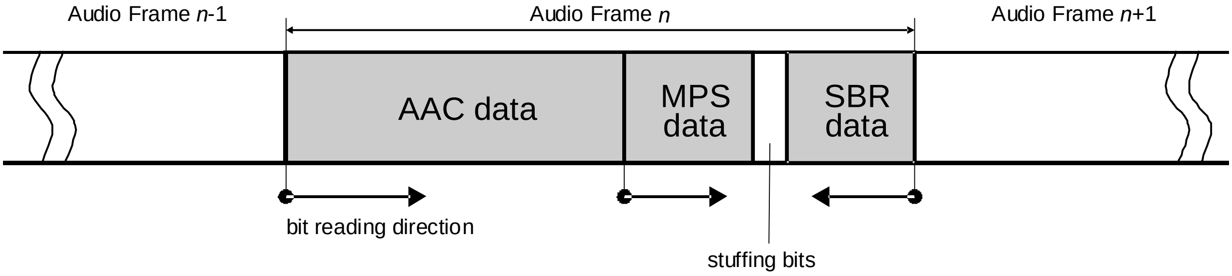


Figure 11: AAC + SBR + MPS frame

The combination of AAC or AAC + SBR with MPEG Surround (MPS) uses the MPEG Surround tool as it is defined in ISO/IEC 23003-1 [9]. The method how to extract the MPS elements from the MPS data and thus the mapping to the format defined in the MPEG standard is given in table 16.

Table 16: Syntax of mps_aac_frame()

Syntax	No. of bits	Note
mps_aac_frame()		
{		
mps_crc_bits;	8	see annex D
ancType;	2	uimsbf
ancStart;	1	uimsbf
ancStop;	1	uimsbf
cnt = ancLenBytes;	8	uimsbf
if (cnt==255) {		
cnt += ancLenBytesAdd;	16	uimsbf
}		
for (i=0; i<cnt; i++) {		
ancDataSegmentByte[i];	8	bslbf
}		
}		

mps_crc_bits Cyclic redundancy checksum for the MPS bit stream part. The CRC algorithm covers all bits in the mps_aac_frame() element excluding the mps_crc_bits itself.

ancType Indicates type of ancillary data, see table 17.

Table 17: Definition of ancType

ancType	Meaning
0x0	SacDataFrame(0) (MPEG Surround frame)
0x1	SacDataFrame(1) (MPEG Surround header and MPEG Surround frame)
0x2, 0x3	(reserved)
NOTE: SacDataFrame() is defined in [9].	

ancStart Indicates if data segment begins a data block.

ancStop Indicates if data segment ends a data block.

ancLenBytes Number of bytes in data segment.

ancLenBytesAdd Additional number of bytes in data segment, needed if the data segments contains 255 or more bytes.

ancDataSegmentByte The concatenation of all ancDataSegmentByte from consecutive mps_aac_frame(), starting from the mps_aac_frame() with ancStart==1 up to and including the mps_aac_frame() with ancStop==1 forms one data block. In case a complete data block is contained in one mps_aac_frame(), it has ancStart==1 and ancStop==1. If ancType==0x0 or ancType==0x1 then this data block constitutes one SacDataFrame() syntax element, padded at the end to obtain an integer number of bytes.

6 Multiplex definition

6.1 Introduction

The DRM transmission super frame consists of three channels: the Main Service Channel (MSC), the Fast Access Channel (FAC) and the Service Description Channel (SDC). The MSC contains the data for the services. The FAC provides information on the channel width and other such parameters and also provides service selection information to allow for fast scanning. The SDC gives information on how to decode the MSC, how to find alternative sources of the same data, and gives the attributes of the services within the multiplex. It can include links to analogue simulcast services.

6.2 Main Service Channel (MSC)

6.2.1 Introduction

The Main Service Channel (MSC) contains the data for all the services contained in the DRM multiplex. The multiplex may contain between one and four services, and each service may be either audio or data. The gross bit rate of the MSC is dependent upon the DRM channel bandwidth and the transmission mode. The MSC can be encoded using either one or two protection levels.

6.2.2 Structure

The MSC contains between one and four streams. Each stream is divided into logical frames carrying content for a fixed period of time. Audio streams comprise compressed audio and optionally they can carry text messages. Data streams may be composed of data packets, carrying information for up to four "sub-streams". An audio service comprises one audio stream and optionally one to four data streams or data sub-streams. A data service comprises one data stream or data sub-stream.

For robustness modes A, B, C and D, the logical frames are each 400 ms long. If the stream carries audio, the logical frame carries the data for one audio super frame.

For robustness mode E, the logical frames are each 100 ms long. If the stream carries audio, the logical frame carries the data for either the first or the second part of one audio super frame containing the audio information for 200 ms duration.

The logical frames from all the streams are mapped together to form multiplex frames of the same duration, which are passed to the channel coder.

The multiplex configuration is signalled using the SDC.

Annex M contains some examples of different MSC configurations.

6.2.3 Building the MSC

The MSC consists of a sequence of multiplex frames. The multiplex frames are passed to the channel coder.

The multiplex frames are built by placing the logical frames from each stream together, taking into account the protection level assigned to each stream. When two protection levels are used for the MSC, the payload of streams using the higher protection level is carried in part A of the multiplex frame and the payload of streams using the lower protection level is carried in part B of the multiplex frame. When a single protection level is used for the MSC, all the payload is carried in part B of the multiplex frame.

The multiplex frame is constructed by taking the data from the logical frame from the lowest numbered stream that uses the higher protection level (if used) and placing it at the start of the multiplex frame. Next the data from the logical frame from the next lowest numbered stream that uses the higher protection level is appended and so on until all the higher protected streams have been transferred. Then the data from the logical frame from the lowest numbered stream that uses the lower protection level is appended, followed by the data from the next lowest numbered stream, and so on until all the lower protected streams have been transferred. The higher protected part of the multiplex frame is designated part A and the lower protected part is designated part B in the multiplex description. The combination of the protection levels and data lengths for each stream forms its UEP profile. An example is given in annex C.

The capacity of the multiplex frame is larger than or equal to the sum of the logical frames from which it is formed. The remainder, if any, of the multiplex frame shall be filled with 0s. These bits shall be ignored by the receiver.

NOTE: No padding bits are inserted between the end of part A and the beginning of part B. The capacity of part A of the multiplex frame is equal to the sum of the higher protected parts of the logical frames, but as a result of restrictions introduced by the channel encoding procedure applied for DRM (see clause 7.2.1.1), some of the bits nominally belonging to the lower protected part B of a multiplex frame might in fact be protected at the higher level.

6.2.4 Reconfiguration

The multiplex may be reconfigured at transmission super frame boundaries.

A reconfiguration of the multiplex occurs when the channel parameters in the FAC are changed, or when the services in the multiplex are reorganized. The new configuration is signalled ahead of time in the SDC and the timing is indicated by the reconfiguration index in the FAC. Clause 6.4.6 describes the signalling of a reconfiguration.

6.3 Fast Access Channel (FAC)

6.3.1 Introduction

The FAC is used to provide information on the channel parameters required for the de-modulation of the multiplex as well as basic service selection information for fast scanning and an indicator of Warning/Alarm announcements.

The channel parameters (for example the spectrum occupancy and interleaving depth) allow a receiver to begin to decode the multiplex effectively. It also contains information about the services in the multiplex to allow the receiver to either decode this multiplex or change frequency and search again.

6.3.2 Structure

Each transmission frame contains an FAC block. An FAC block contains parameters that describe the channel and parameters to describe either one or two services along with a CRC.

For robustness modes A, B, C and D, one set of service parameters shall be transmitted and for robustness mode E, two sets of service parameters shall be transmitted. When more services are carried in the multiplex than can be described within one FAC block, a number of FAC blocks are required to describe all the services, see clause 6.3.6.

6.3.3 Channel parameters

The channel parameters are as follows:

- Base/Enhancement flag 1 bit
- Identity 2 bits
- RM flag 1 bit
- Spectrum occupancy 3 bits
- Interleaver depth flag 1 bit
- MSC mode 2 bits

- SDC mode 1 bit
- Number of services 4 bits
- Reconfiguration index 3 bits
- Toggle flag 1 bit
- rfu 1 bit

The following definitions apply:

Base/Enhancement flag: this 1-bit flag indicates whether the transmission is the base or enhancement layer as follows:

0: Base layer - decodable by all DRM receivers.

1: Enhancement layer - only decodable by receivers with enhancement layer capabilities.

Identity: this 2-bit field identifies the current FAC block within the transmission super frame and also validates the SDC AFS index (see clause 6.4) as follows:

00: first FAC block of the transmission super frame and SDC AFS index is valid.

01: intermediate FAC block of the transmission super frame.

10: last FAC block of the transmission super frame.

11: first FAC block of the transmission super frame and SDC AFS index is invalid.

NOTE: Either one or two intermediate FAC blocks may be present within a transmission super frame depending on the value of the RM flag. Intermediate FAC blocks can be distinguished by using the Toggle flag.

RM flag: this 1-bit field indicates the robustness mode as follows:

0: Robustness modes A, B, C or D; FAC block contains one set of service parameters.

1: Robustness mode E; FAC block contains two sets of service parameters.

The interpretation of the Spectrum occupancy, Interleaver depth flag, MSC mode, SDC mode and Toggle flag parameters are dependent on the value of the RM flag.

Spectrum occupancy: this 3-bit field, coded as an unsigned integer, specifies the nominal channel bandwidth and configuration of the digital signal as follows. See also clause 8.

RM flag = 0:

0: 4,5 kHz.

1: 5 kHz.

2: 9 kHz.

3: 10 kHz.

4: 18 kHz.

5: 20 kHz.

other values reserved.

RM flag = 1:

0: 100 kHz.

other values reserved.

Interleaver depth flag: this 1-bit flag indicates the depth of the time interleaving as follows:

RM flag = 0:

0: 2 s (long interleaving).

1: 400 ms (short interleaving).

RM flag = 1:

0: 600 ms.

1: reserved.

MSC mode: this 2-bit field indicates the modulation mode in use for the MSC as follows:

RM flag = 0:

00: 64-QAM.

01: reserved.

10: reserved.

11: 16-QAM.

RM flag = 1:

00: 16-QAM.

01: reserved.

10: reserved.

11: 4-QAM.

SDC mode: this 1-bit field indicates the modulation mode and code rate in use for the SDC as follows:

RM flag = 0:

0: 16-QAM, code rate = 0,5.

1: 4-QAM, code rate = 0,5.

RM flag = 1:

0: 4-QAM, code rate = 0,5.

1: 4-QAM, code rate = 0,25.

Number of services: this 4-bit field indicates the number of audio and data services as follows:

0000: 4 audio services.

0001: 1 data service.

0010: 2 data services.

0011: 3 data services.

0100: 1 audio service.

0101: 1 audio service and 1 data service.

0110: 1 audio service and 2 data services.

0111: 1 audio service and 3 data services.

1000: 2 audio services.

1001: 2 audio services and 1 data service.

1010: 2 audio services and 2 data services.

1011: reserved.

1100: 3 audio services.

1101: 3 audio services and 1 data service.

1110: reserved.

1111: 4 data services.

Reconfiguration index: this 3-bit field indicates the status and timing of a multiplex reconfiguration. A non-zero value indicates the number of transmission super frames of the old configuration that are transmitted before the new configuration takes effect, see clause 6.4.6.

Toggle flag: this 1-bit flag shall be used to indicate that this transmission frame may contain the start of an audio super frame as follows:

RM flag = 0:

Toggle flag is fixed to zero.

RM flag = 1:

The toggle flag is set to zero for the first and third FAC block of the transmission super frame and to one for the second and fourth FAC block. It may be used in combination with the Identity parameter to distinguish the received transmission frames.

rfu: this 1 bit flag is reserved for future use of the whole FAC parameter definitions and shall be set to zero until defined.

6.3.4 Service parameters

The service parameters are as follows:

- Service identifier 24 bits
- Short Id 2 bits
- Audio CA indication 1 bit
- Language 4 bits
- Audio/Data flag 1 bit
- Service descriptor 5 bits
- Data CA indication 1 bit
- rfa 6 bits

The following definitions apply:

Service identifier: this 24-bit field indicates the unique identifier for this service. For recommendations on setting the Service identifier refer to annex T.

Short Id: this 2-bit field indicates the short identifier assigned to this service and used as a reference in the SDC. The Short Id is assigned for the duration of the service and is maintained through multiplex reconfigurations.

Audio CA indication: this 1-bit flag indicates whether the service uses conditional access as follows:

0: No CA system is used for the audio stream (or the service has no audio stream).

1: CA system is used for the audio stream.

NOTE 1: The details are provided by the SDC data entity type 2.

Every DRM receiver shall check the "Audio CA indication" bit before presenting the audio stream of the audio service. A non-CA capable DRM receiver shall not try to decode the audio stream if the "Audio CA indication" is set to 1.

Language: this 4-bit field indicates the language of the target audience as defined in table 18.

NOTE 2: Further languages are also indicated by SDC data entity type 12.

Audio/Data flag: this 1-bit flag indicates whether the service is audio or data as follows:

0: Audio service.

1: Data service.

Service descriptor: this 5-bit field usually depends upon the value of the Audio/Data flag as follows:

0: Programme type.

1: Application identifier.

Regardless of the value of the Audio/Data flag, two special values are defined:

- the value 30 indicates that a Warning/Alarm announcement is currently active for at least one of the services carried in this multiplex;

NOTE 3: This is to allow for low power receiver implementations to recognize an active Warning/Alarm announcement before full decoding of the SDC is completed, see annex N.

- the value 31 indicates that a standard DRM receiver should skip this multiplex and continue to scan for services.

NOTE 4: This is to allow for engineering test transmissions to be ignored by standard receivers.

Programme type: this 5-bit field indicates the programme type of an audio service as defined in table 19.

Application identifier: this 5-bit field indicates the application identifier of a data service as defined in ETSI TS 101 968 [7].

Data CA indication: this 1-bit flag indicates whether the service uses conditional access as follows:

0: No CA system is used for the data stream/sub-stream (or the service has no data stream/sub-stream).

1: CA system is used for the data stream/sub-stream.

NOTE 5: The details are provided by the SDC data entity type 2.

Every DRM receiver shall check the "Data CA indication" bit before presenting the data stream/sub-stream of the audio or data service. A non-CA capable DRM receiver shall not try to decode the data stream/sub-stream if the "Data CA indication" is set to 1.

rfa: these 6 bits are reserved for future additions and shall be set to zero until defined.

Table 18: Language codes

Decimal number	Language	Decimal number	Language
0	No language specified	8	Hindi
1	Arabic	9	Japanese
2	Bengali	10	Javanese
3	Chinese (Mandarin)	11	Korean
4	Dutch	12	Portuguese
5	English	13	Russian
6	French	14	Spanish
7	German	15	Other language

Table 19: Programme type codes

Decimal number	Programme type	Decimal number	Programme type
0	No programme type	16	Weather/meteorology
1	News	17	Finance/Business
2	Current Affairs	18	Children's programmes
3	Information	19	Social Affairs
4	Sport	20	Religion
5	Education	21	Phone In
6	Drama	22	Travel
7	Culture	23	Leisure
8	Science	24	Jazz Music
9	Varied	25	Country Music
10	Pop Music	26	National Music
11	Rock Music	27	Oldies Music
12	Easy Listening Music	28	Folk Music
13	Light Classical	29	Documentary
14	Serious Classical	30	Not used - warning/alarm indicator
15	Other Music	31	Not used - skip indicator
NOTE: Codes 30 and 31 are not used as programme types as they are special indicators: they are not displayed to users.			

6.3.5 CRC

The 8-bit Cyclic Redundancy Check shall be calculated on the channel and service parameters. It shall use the generator polynomial $G_8(x) = x^8 + x^4 + x^3 + x^2 + 1$. See annex D.

When the RM flag = 0, the CRC is calculated over 64-bits formed by concatenating the 20-bits of channel parameters and the 44-bits of service parameters. When the RM flag = 1, the CRC is calculated over 112-bits formed by concatenating the 20-bits of channel parameters, the 88-bits of service parameters (2 sets of 44-bits) and 4-bits set to zero. These 4-bits are used to calculate the CRC but are not forwarded for coding and transmission.

6.3.6 FAC repetition

The FAC channel parameters shall be sent in each FAC block. The FAC service parameters for one or two services shall be sent in each FAC block. When more than one FAC block is needed to signal all the services in the multiplex, the repetition pattern is significant to the receiver scan time. When all services are of the same type (e.g. all audio or all data) then the services shall be signalled sequentially. When a mixture of audio and data services is present then the patterns shown in table 20 shall be signalled. In the case when there is only one service and the FAC block signals two sets of service parameters, both sets shall contain identical content.

Table 20: Service parameter repetition patterns for mixtures of audio and data services

Number of audio services	Number of data services	Repetition pattern when FAC block contains one set of service parameters	Repetition pattern when FAC block contains two sets of service parameters
1	1	A1A1A1A1D1	A1D1
1	2	A1A1A1A1D1A1A1A1A1D2	A1D1 A1D2
1	3	A1A1A1A1D1A1A1A1A1D2A1A1A1A1D3	A1D1 A1D2 A1D3
2	1	A1A2A1A2D1	A1A2 D1A1 A2D1
2	2	A1A2A1A2D1A1A2A1A2D2	A1A2 A1D1 A2D2
3	1	A1A2A3A1A2A3D1	A1A2 A3D1

Where A*n* designates an audio service and D*n* designates a data service.

6.4 Service Description Channel (SDC)

6.4.1 Introduction

This clause describes the format and content of the SDC. The SDC gives information on how to decode the MSC, how to find alternative sources of the same data, and gives attributes to the services within the multiplex.

The data capacity of the SDC varies with the spectrum occupancy of the multiplex and other parameters. The SDC capacity can also be increased by making use of the AFS index.

Alternative frequency checking may be achieved, without loss of service, by keeping the data carried in the SDC quasi-static. Therefore, the data in the SDC frames has to be carefully managed.

6.4.2 Structure

An SDC block is the SDC data contained in one transmission super frame.

The SDC is treated as a single data channel. The total amount of data to be sent may require more than a single SDC block to send. An AFS index is therefore provided to permit a receiver to know when the next occurrence of the current SDC block will be transmitted, and so allow for alternative frequency checking and switching (AFS). A validity function is provided in the FAC to indicate whether the AFS index is valid or not, indicating to a receiver when the AFS function can operate.

The SDC block is made up as follows:

- AFS index 4 bits.
- data field *n* bytes.
- CRC 16 bits.
- padding *k* bits.

The **AFS index** is an unsigned binary number in the range 0 to 15 that indicates the number of transmission super frames which separate this SDC block from the next with identical content when the identity field in the FAC is set to 00. The AFS index shall be identical for all SDC blocks. The AFS index may be changed at reconfiguration.

The **data field** carries a variable number of data entities. It may contain padding. The length of the data field depends upon the robustness mode, SDC mode and spectrum occupancy, and is given in table 21.

Table 21: Length of SDC data field

Robustness mode	SDC mode	Length of data field (bytes)					
		Spectrum occupancy					
		0	1	2	3	4	5
A	0	37	43	85	97	184	207
	1	17	20	41	47	91	102
B	0	28	33	66	76	143	161
	1	13	15	32	37	70	79
C	0	-	-	-	68	-	147
	1	-	-	-	32	-	72
D	0	-	-	-	33	-	78
	1	-	-	-	15	-	38
E	0	113	-	-	-	-	-
	1	55	-	-	-	-	-

The **CRC** (Cyclic Redundancy Check) field shall contain a 16-bit CRC calculated over the AFS index coded in an 8-bit field (4 msbs are 0) and the data field. It shall use the generator polynomial $G_{16}(x) = x^{16} + x^{12} + x^5 + 1$. See annex D.

The **padding** field contains 0 bits to 7 bits to complete the transmission super frame. The value of *k* depends on the robustness mode, SDC mode and spectrum occupancy. The padding bits shall be set to zero. These bits shall be ignored by the receiver.

6.4.3 Data entities

6.4.3.0 Introduction

The data field is filled with data entities. Every data entity has a 12-bit header and a variable length body. The header has the following format:

- length of body 7 bits.
- version flag 1 bit.
- data entity type 4 bits.

The following definitions apply:

The **length of body** gives the number of whole bytes occupied by the data entity body.

The **version flag** controls the management of data in the receiver.

The **data entity type** is a number that determines the identity of the data entity.

The version flag allows three different mechanisms to control data management in the receiver, as specified below. The actual mechanism used is specified for each data entity.

Reconfiguration: For data entities using this mechanism, the version flag indicates whether the data is for the current (= 0) or next (= 1) configuration.

List: For data entities using this mechanism, the version flag indicates the version of the list. When any of the data in the list changes, the flag is inverted and the existing data in the receiver is discarded. The version flag applies to all the data delivered using the data entity type.

Unique: For data entities using this mechanism, the version flag has no meaning and shall be set to 0. These data entities carry data that is unique and therefore do not require any change mechanism.

The body of the data entities shall be at least 4 bits long. The length of the body, excluding the initial 4 bits, shall be signalled by the header.

If there is space remaining in the data field, it shall be filled with padding. The padding bytes shall take the value 0x00.

6.4.3.1 Multiplex description data entity - type 0

Each SDC block should contain a multiplex description entity. This data entity uses the reconfiguration mechanism for the version flag. The current configuration can always be signalled. During a reconfiguration (i.e. when the FAC reconfiguration index is non-zero) the next configuration shall be signalled. This data entity describes the multiplex of streams within the MSC and the UEP profile of each stream. The information is as follows:

- protection level for part A 2 bits.
- protection level for part B 2 bits.
- stream description for stream 0 24 bits.

and optionally, dependent upon the number of streams in the multiplex:

- stream description for stream 1 24 bits.
- stream description for stream 2 24 bits.
- stream description for stream 3 24 bits.

The stream descriptions for stream 0, and, when present, for streams 1, 2 and 3, are as follows:

- data length for part A 12 bits.
- data length for part B 12 bits.

The following definitions apply:

protection level for part A: this field gives the overall coding rate for data in part A (see clause 7.5.1).

protection level for part B: this field gives the overall coding rate for data in part B (see clause 7.5.1).

data length for part A: this field gives the net length of data in bytes in part A of the logical frame used by this stream.

data length for part B: this field gives the net length of data in bytes in part B of the logical frame used by this stream.

When EEP is allocated to the multiplex frame (i.e. only one protection level is used for the MSC) then the data length for the part A fields shall be set to 0 and the protection level for part A fields shall be set to 0.

When UEP is allocated to the multiplex frame then part A is the higher protected part and part B is the lower protected part.

NOTE 1: If more than one service is carried in the multiplex, a service may be carried in either part (part A or part B). In this way, different services can be transported using equal error protection at the higher level or equal error protection at the lower level in the same multiplex.

NOTE 2: The receiver may determine the number of streams present in the multiplex by dividing the value of the length field of the header by three.

6.4.3.2 Label data entity - type 1

Services should be labelled. The label should be sent in every SDC block to enable fast display, although for data services the repetition rate can be lowered. This data entity uses the unique mechanism for the version flag. The information is as follows:

- Short Id 2 bits
- rfu 2 bits
- label *n* bytes

The following definitions apply:

Short Id: this field contains the short Id that relates the information to the Service Id provided by the FAC.

rfu: these 2 bits are reserved for future use of the remainder of the parameter field and shall be set to zero until defined.

label: this is a variable length field of up to 64 bytes containing character data for up to 16 displayable characters using UTF-8 coding (ISO/IEC 10646 [3]).

NOTE 1: The length of the **label** field (in bytes) is given by the **length of body** field of the data entity header.

Every **label** shall include at least one displayable character. Only base characters contribute to the maximum length of 16 displayable characters (although the maximum **label** length of 64 bytes shall be respected).

The default direction of the **label** is Left-To-Right (LTR). When Right-To-Left (RTL) scripts are used, or contextual or combining characters are included in the **label**, additional signalling is provided in form of the 4-bit **text control** field (see clause 6.7.2).

If any bit of the **text control** field is set to 1, then it shall be carried as the first byte of the **label** field, with the most significant 4 bits set to zero.

For backwards compatibility reasons, the byte containing the **text control** field is not inserted into the **label** field if all bits are equal to 0. This is the case for uni-directional Left-To-Right (LTR) labels without contextual or combining characters.

NOTE 2: Since the byte carrying the **text control** field will have values from 0x01 to 0x0F, which correspond to Unicode control codes, it is most likely that legacy receivers that do not support the **text control** field mechanism will treat this byte as a non-displayable character.

6.4.3.3 Conditional access parameters data entity - type 2

This data entity allows the conditional access parameters to be sent. This data entity uses the reconfiguration mechanism for the version flag.

- Short Id 2 bits
- Audio CA flag 1 bit
- Data CA flag 1 bit
- CA system specific information *n* bytes

The following definitions apply:

Short Id: this field contains the short Id that relates the information to the Service Id provided by the FAC.

Audio CA flag: this 1-bit flag indicates whether the conditional access parameters refer to an audio stream as follows:

0: Parameters do not refer to an audio stream.

1: Parameters refer to an audio stream.

NOTE 1: In case of a data service this flag will be 0.

Data CA flag: this 1-bit flag indicates whether the conditional access parameters refer to a data stream/sub-stream as follows:

0: Parameters do not refer to a data stream/sub-stream.

1: Parameters refer to a data stream/sub-stream.

NOTE 2: In case of an audio service that does not have a data stream/sub-stream this flag will be 0.

NOTE 3: In case of an audio service with data stream(s)/sub-stream(s) all use the same CA system.

CA system specific information: this is a variable length field containing CA system specific data.

NOTE 4: The CA system specific information should include a proprietary CA system/version identifier along with a system specific ID and/or checksum, to allow each CA decoder to identify its CA configuration data and to reject CA configuration data belonging to other CA systems.

NOTE 5: An audio service can have a scrambled audio stream and one or more scrambled data streams/sub-streams and the conditional access parameters can be different for audio and data. In this case two *Conditional access parameters data entity - type 2* are needed. If the audio stream and the data streams/sub-streams use identical conditional access parameters then one *Conditional access parameters data entity - type 2* is sufficient; both the Audio CA flag and the Data CA flag are set to 1.

6.4.3.4 Alternative frequency signalling: Multiple frequency network information data entity - type 3

This data entity is used to provide receivers with information about the DRM frequencies on which:

- the whole multiplex of services can be found;
- some of the services of the multiplex can be found;
- the base/enhancement layer of the whole multiplex of services can be found;
- the base/enhancement layer of some of the services of the multiplex can be found.

It is also used to provide receivers with information as to whether the frequencies can be used for seamless alternate frequency checking and switching, see annex G.

This data entity uses the list mechanism for the version flag.

NOTE 1: If the same audio programme or data application is broadcast using a different DRM Service identifier, or if it is broadcast using another broadcast system, then data entity type 11 is used to indicate the frequency.

The frequencies may be restricted to certain times and/or geographic areas in combination with data entities type 4, 7 and 13 respectively.

The information is as follows:

- Synchronous Multiplex flag 1 bit
- Layer flag 1 bit
- Service Restriction flag 1 bit
- Region/Schedule flag 1 bit
- Service Restriction field 0 or 8 bits
- Region/Schedule field 0 or 8 bits
- *n* frequencies $n \times 16$ bits

The following definitions apply:

Synchronous Multiplex flag: this flag indicates whether the multiplex is broadcast synchronously as follows:

0: multiplex is not synchronous (different content and/or channel parameters and/or multiplex parameters and/or signal timing in target area).

1: multiplex is synchronous (identical content and channel parameters and multiplex parameters and signal timing in target area).

Layer flag: this flag indicates whether the frequencies given apply to the base layer of the DRM multiplex or to the enhancement layer as follows:

0: base layer.

1: enhancement layer.

Service Restriction flag: this flag indicates whether all or just some of the services of the tuned multiplex are available in the DRM multiplex on the frequencies given as follows:

0: all services in the tuned multiplex are available on the frequencies given.

1: a restricted set of services are available on the frequencies given.

The following combinations of these three flags are defined in table 22.

Table 22: Type 3 flag combinations

Synchronous Multiplex flag	Layer flag	Service Restriction flag	Usage
0	0	0	non-synchronous multiplex with same service list
0	0	1	non-synchronous multiplex with restricted service list
0	1	0	not used
0	1	1	not used
1	0	0	synchronous multiplex with same service list (base layer) (see note)
1	0	1	not used
1	1	0	synchronous enhancement layer multiplex with same service list
1	1	1	synchronous enhancement layer multiplex with restricted service list
NOTE: This combination is always used for indicating the frequencies of the base layer from the enhancement layer, even if the base layer contains *more* services than the enhancement layer.			

Region/Schedule flag: this field indicates whether the list of frequencies is restricted by region and/or schedule or not as follows:

0: no restriction.

1: region and/or schedule applies to this list of frequencies.

Service Restriction field: this 8 bit field is only present if the Service Restriction flag is set to 1. The information is as follows:

- Short Id flags 4 bits
- rfa 4 bits

Short Id flags: this 4 bit field indicates, which services (identified by their Short Id) of the tuned DRM multiplex are carried in the DRM multiplex on the alternative frequencies by setting the corresponding bit to 1. The first bit (msb) refers to Short Id 3, while the last bit (lsb) refers to Short Id 0 of the tuned DRM multiplex.

Region/Schedule field: this 8 bit field is only present if the Region/Schedule flag is set to 1. The information is as follows:

- Region Id 4 bits
- Schedule Id 4 bits

Region Id: this field indicates whether the region is unspecified (value 0) or whether the alternative frequencies are valid just in certain geographic areas, in which case it carries the Region Id (value 1 to 15). The region may be described by one or more "Alternative frequency signalling: Region definition data entity - type 7" and/or "Alternative frequency signalling: detailed region definition data entity - type 13" with this Region Id.

Schedule Id: this field indicates whether the schedule is unspecified (value 0) or whether the alternative frequencies are valid just at certain times, in which case it carries the Schedule Id (value 1 to 15). The schedule is described by one or more "Alternative frequency signalling: Schedule definition data entity - type 4" with this Schedule Id.

***n* frequencies:** this field carries *n* 16 bit fields. *n* is in the range 1 to 16. Each 16 bit field contains the following information:

- multiplier 1 bit
- frequency value 15 bits

NOTE 2: The number of frequencies, *n*, is determined from the length field of the header and the value of the Service Restriction flag and the Region/Schedule flag.

multiplier: this 1-bit field shall indicate the frequency multiplier as follows:

0: 1 (the frequencies field can indicate from 0 to 32 767 kHz in 1 kHz steps; indicated transmission has robustness mode A, B, C or D).

1: 10 (the frequencies field can indicate from 0 to 327 670 kHz in 10 kHz steps; indicated transmission has robustness mode E).

frequency value: this 15 bit field is coded as an unsigned integer and gives the frequency in multiples of 1 or 10 kHz, depending on the value of the multiplier field.

rfa: these 1-bit or 4-bit fields (if present) are reserved for future additions and shall be set to zero until defined.

Additional information on Alternative Frequency Signalling is provided in annexes F and G.

6.4.3.5 Alternative frequency signalling: Schedule definition data entity - type 4

This entity allows a frequency schedule to be transmitted. This data entity uses the list mechanism for the version flag. This information is as follows:

- Schedule Id 4 bits
- Day Code 7 bits
- Start Time 11 bits
- Duration 14 bits

The following definitions apply:

Schedule Id: this field indicates the Schedule Id for the defined schedule. Up to 15 different schedules with an individual Schedule Id (values 1 to 15) can be defined; the value 0 shall not be used, since it indicates "unspecified schedule" in data entity type 3 and 11.

Day Code: this field indicates which days the frequency schedule (the following Start Time and Duration) applies to. The msb indicates Monday, the lsb Sunday. Between one and seven bits may be set to 1.

Start Time: this field indicates the time from when the frequency is valid. The time is expressed in minutes since midnight UTC. Valid values range from 0 to 1 439 (representing 00:00 to 23:59).

Duration: this field indicates how long the frequency is valid starting from the indicated Start Time. The time is expressed in minutes. Valid values range from 1 to 16 383.

NOTE: The Duration may signal a time interval of more than one week. See annex O for interpretation rules and examples.

Additional information on Alternative Frequency Signalling is provided in annexes F and G.

6.4.3.6 Application information data entity - type 5

All data services (or data applications for audio services) are described by this data entity. Additional information regarding the handling of data services is given in ETSI TS 101 968 [7].

Many applications may require additional data to describe them that is specific to that application. This data entity uses the reconfiguration mechanism for the version flag. The content is described by the appropriate application specification. The general form of the entity is as follows:

- Short Id 2 bits
- Stream Id 2 bits
- Packet mode indicator 1 bit
- descriptor 7 bits or 15 bits
- application data *n* bytes

The following definitions apply:

Short Id: this field indicates the short Id for the service concerned.

Stream Id: this field indicates the stream Id of the stream which carries the data service (or data application) concerned.

Packet mode indicator: this field indicates whether the service is carried in packet mode or not as follows:

0: synchronous stream mode.

1: packet mode.

NOTE 1: All data services (or data applications) contained in one data stream signal the same Packet mode indicator value.

descriptor: the format of this field depends upon the value of the Packet mode indicator field as follows:

when **Packet mode indicator** = 0:

- rfa 3 bits
- enhancement flag 1 bit
- application domain 3 bits

rfa: these 3 bits are reserved for future additions and shall be set to zero until defined.

enhancement flag: this field indicates whether enhancement data is available in another channel as follows:

0: no enhancement available.

1: enhancement is available.

application domain: this field indicates the source of the data application specification. The interpretation of this field is given in ETSI TS 101 968 [7].

when **Packet mode indicator** = 1:

- data unit indicator 1 bit
- packet Id 2 bits
- enhancement flag 1 bit
- application domain 3 bits
- packet length 8 bits

data unit indicator: this field indicates whether the data stream is composed of single packets or data units as follows:

0: single packets.

1: data units.

packet Id: this two-bit field, coded as unsigned integer, indicates the Packet Id carried in the header of packets intended for this service. When FEC is added to a packet mode stream (see clause 6.6.5), packet Id = 3 is reserved for transporting error correction information for the whole packet mode data stream (and optionally, padding packets).

enhancement flag: this field indicates whether enhancement data is available in another channel as follows:

0: no enhancement available.

1: enhancement is available.

application domain: this field indicates the source of the data application specification. The interpretation of this field is given in ETSI TS 101 968 [7].

packet length: this field, coded as an unsigned integer in the range 1 to 255, indicates the length in bytes of the data field of each packet (the total packet length is three bytes longer as it includes the header and CRC fields).

NOTE 2: All packets contained in one data stream have the same length (see clause 6.6.4).

application data: this field of variable length is defined by the data service (or data application) specification. The interpretation of this field is given in ETSI TS 101 968 [7].

6.4.3.7 Announcement support and switching data entity - type 6

This data entity indicates which types of announcements are supported by the tuned multiplex or by another DRM multiplex/another broadcast system. It also indicates which of the services of the tuned multiplex should switch in case of an active announcement.

NOTE: Announcement switching flags can only be signalled for those announcement types whose corresponding Announcement support flags are currently set as active.

This data entity uses the list mechanism for the version flag. The version flag status shall remain the same if only the "Announcement switching flags" are changed. The version flag shall be inverted in case of changes to any other announcement information.

The following information is necessary:

- Short Id flags 4 bits
- Same Multiplex/Other Service flag 1 bit
- Short Id/Announcement Id 2 bits
- rfa 1 bit
- Announcement support flags 10 bits
- Announcement switching flags 10 bits

The following definitions apply:

Short Id flags: this 4 bit field indicates to which services (identified by their Short Id) of the tuned DRM multiplex the announcement definition applies to by setting the corresponding bit to 1. The first bit (msb) refers to Short Id 3, while the last bit (lsb) refers to Short Id 0 of the tuned DRM multiplex.

Same Multiplex/Other Service flag: this field indicates if the announcements are carried in the tuned DRM multiplex or not, as follows:

0: announcements are carried in the tuned multiplex.

1: announcements are carried elsewhere.

Short Id/Announcement Id: the content of this 2 bit field depends upon the value of the Same Multiplex/Other Service flag as follows:

Same Multiplex/Other Service flag = 0:

- Short Id 2 bits

Same Multiplex/Other Service flag = 1:

- Announcement Id 2 bits

Short Id: this field signals the Short Id of the service within the tuned DRM multiplex which carries the announcement content.

Announcement Id: this field carries the Announcement Id (value 0 to 3). The Other Service Ids (and potentially frequencies) carrying the programme with the announcement content are described by one or multiple "Alternative frequency signalling: Other services data entity - type 11" (with the "Short Id/Announcement Id flag" bit set to 1, indicating that the Other Service Id and/or the list of frequencies belong to a programme carrying announcement content).

rfa: this 1 bit is reserved for future additions and shall be set to zero until defined.

Announcement support flags: this 10-bit field specifies the types of announcements that are described by this data entity and provided either by one service in the tuned DRM multiplex or by another service on another frequency as follows:

B_i (i = 0 to 9).

0: Announcement type not provided.

1: Announcement type provided.

The meaning of each bit is as follows:

b_0: Travel.

b_1: News flash.

b_2: Weather flash.

b_3: Warning/Alarm.

b_4: Warning/Alarm Test.

b_5 to b_9: Reserved for future definition.

Announcement switching flags: the individual bits of this 10-bit field indicate whether or not a particular announcement type is currently active. The flags are coded as follows:

B_i (i = 0 to 9).

0: Announcement type not valid (currently not active).

1: Announcement type valid (currently active).

The meaning of each bit is as defined for the Announcement support flags above.

Special rules apply to the use of the announcement flags b_3 and b_4 which are used for signalling and testing emergency Warning/Alarm announcements respectively. Announcement flag b_3 shall only be used by competent authorities.

When the announcement flag b_3 (Warning/Alarm) is active (set to 1) in the **Announcement switching flags**, this status shall be reflected in the FAC by setting the **Service description** field of all service parameter blocks to the value 30, see clause 6.3.4 and annex N.

In order to ensure that the whole broadcast chain delivering a Warning/Alarm announcement may be tested without causing unwanted listener confusion, the Warning/Alarm Test announcement flag b_4 is provided. It allows a standardized way for public authorities to test the Warning/Alarm announcement system without signalling an actual Warning/Alarm: testing with an actual Warning/Alarm announcement would cause unacceptable disruption and could lead to public unease. The test bit provides the necessary function to test the emergency procedures and emergency systems in a controlled way.

Support of Warning/Alarm Test announcements is optional in consumer receivers: if it is implemented it shall be explicitly enabled by the user and automatically disabled after a Warning/Alarm Test announcement has ceased. Special receivers designed (e.g. for use by public authorities) may behave differently.

Additional information on announcements is provided in annex F.

6.4.3.8 Alternative frequency signalling: Region definition data entity - type 7

This data entity allows the definition of geographic areas for which a set of alternative frequencies is provided. This data entity uses the list mechanism for the version flag.

A region can be specified as a geographical area using longitude/latitude and extent values. The area is defined in terms of multiples of 1 × 1 degree "squares". It therefore gives a resolution of (EW × NS) 111 km × 111 km (at equator) or 31 km × 111 km at 70° latitude (e.g. Scandinavia, Canada). The coding provided allows for the signalling of squares of at least about 8 000 km × 14 000 km for < 73° latitude.

The area may in addition be defined in terms of CIRAF zones.

If both geographical areas and CIRAF zones are defined per Region Id and can be evaluated by a receiver, the intersection region shall be used.

This information is as follows:

- Region Id 4 bits
- Latitude 8 bits
- Longitude 9 bits
- Latitude Extent 7 bits
- Longitude Extent 8 bits

and optionally:

- *n* CIRAF Zones *n* × 8 bits

The following definitions apply:

Region Id: this field indicates the identifier for this region definition. Up to 15 different geographic regions with an individual Region Id (values 1 to 15) can be defined; the value 0 shall not be used, since it indicates "unspecified geographic area" in data entity type 3 and 11. The Region Id values are used in common for SDC data entities 7 and 13; area definitions per Region Id can be specified by any combination of SDC entities 7 and 13 (see table 24 for limitations).

Latitude: this field specifies the southerly point of the area in degrees, as a 2's complement number between -90 (south pole) and +90 (north pole).

Longitude: this field specifies the westerly point of the area in degrees, as a 2's complement number between -180 (west) and +179 (east).

Latitude Extent: this field specifies the size of the area to the north, in 1° steps; the value of Latitude plus the value of Latitude Extent shall be equal or less than 90.

Longitude Extent: this field specifies the size of the area to the east, in 1° steps; the value of Longitude plus the value of Longitude Extent may exceed the value +179 (i.e. wrap into the region of negative longitude values).

***n* CIRAF Zones:** this field, when present, carries *n* CIRAF zones (*n* in the range 0 to 16). Each CIRAF zone is coded as an 8 bit unsigned binary number in the range 1 to 85.

NOTE: The number of CIRAF zones, *n*, is determined from the length field of the header - 4.

To check whether a certain longitude value is inside the specified longitude range, the following formula in pseudo program code shall be used (with my_longitude in the range -180 to +179):

```
inside_area = ( (my_longitude >= longitude) AND
                (my_longitude <= (longitude + longitude_extent) ) OR
              ( ((longitude + longitude_extent) >= +180) AND
                (my_longitude <= (longitude + longitude_extent - 360)) )
```

This data entity can be used in combination with data entity type 13 which defines the geographic regions with greater resolution.

Additional information on Alternative Frequency Signalling is provided in annexes F and G.

6.4.3.9 Time and date information data entity - type 8

The current time and date can be specified to allow a receiver to follow frequency schedules, etc. This data entity uses the unique mechanism for the version flag. The data entity is coded as follows:

- Modified Julian Date 17 bits
- UTC (hours and minutes) 11 bits

  and optionally:
- rfu 2 bits
- Local Time Offset sense 1 bit
- Local Time Offset value 5 bits

The following definitions apply:

Modified Julian Date: this field indicates the date in MJD format.

UTC: this field specifies the current UTC time expressed in hours (5 bits) and minutes (6 bits).

rfu: this 2-bit field is reserved for future use of the Local Time Offset sense and Local Time Offset value fields and shall be set to zero until defined.

Local Time Offset sense: this field, when present, shall indicate the sense of the Local Time Offset (LTO) from UTC, as follows:

0: positive offset, local time is in advance of UTC;

1: negative offset, local time is behind UTC.

Local Time Offset value: this field, when present, shall indicate the value of the Local Time Offset (LTO) from UTC. It is expressed in multiples of half hours. When combined with the Local Time Offset sense it permits the LTO to indicate -15,5 hours to +15,5 hours with respect to UTC.

The presence of the rfu, Local Time Offset sense and Local Time Offset value fields shall be determined from the value of the length field of the SDC data entity header.

When the time and date are signalled, this data entity shall be carried in the first SDC block on or after the minute's edge.

NOTE: It is intended that the LTO is used to indicate the local time in the targeted reception area. If multiple time zones are likely to be present in the targeted reception area, the LTO should not be used.

6.4.3.10 Audio information data entity - type 9

Each audio service needs a detailed description of the parameters needed for audio decoding. This data entity uses the reconfiguration mechanism for the version flag.

- Short Id 2 bits
- Stream Id 2 bits
- audio coding 2 bits
- SBR flag 1 bit
- audio mode 2 bits
- audio sampling rate 3 bits
- text flag 1 bit
- enhancement flag 1 bit
- coder field 5 bits
- rfa 1 bit
- codec specific config 8*n* bits

The following definitions apply:

short Id: this field indicates the short Id for the service concerned.

Stream Id: this field indicates the stream Id of the stream that carries the service concerned.

audio coding: this field indicated the source coding system as follows:

00: AAC.

01: reserved.

10: reserved.

11: xHE-AAC.

SBR flag: this field depends upon the value of the audio coding field as follows:

audio coding field = 00 (AAC):

0: SBR not used.

1: SBR used.

audio coding field = 01 (reserved), 10 (reserved) or 11 (xHE-AAC):

- rfa 1 bit.

audio mode: this field depends upon the value of the audio coding field as follows:

audio coding field = 00 (AAC):

00: mono.

01: parametric stereo.

10: stereo.

11: reserved.

audio coding field = 01 (reserved) or 10 (reserved):

- rfa 2 bits.

audio coding field = 11 (xHE-AAC):

00: mono.

01: reserved.

10: stereo.

11: reserved.

audio sampling rate: this field indicates the audio sampling rate of the core coder and depends upon the value of the audio coding field as follows:

audio coding field = 00 (AAC):

000: reserved.

001: 12 kHz.

010: reserved.

011: 24 kHz.

100: reserved.

101: 48 kHz.

110: reserved.

111: reserved.

audio coding field = 01 (reserved) or 10 (reserved):

- rfa 3 bits.

audio coding field = 11 (xHE-AAC):

000: reserved.

001: reserved.

010: 16 kHz.

011: 19,2 kHz.

100: 24 kHz.

101: 32 kHz.

110: 38,4 kHz.

111: 48 kHz.

text flag: this field indicates whether a text message is present or not as follows:

0: no text message is carried.

1: text message is carried (see clause 6.5).

enhancement flag: this field indicates whether audio enhancement data is available in another channel as follows:

0: no enhancement available.

1: enhancement is available.

coder field: this field depends upon the value of the audio coding field and SBR flag as follows:

audio coding field = 00 (AAC) or 11 (xHE-AAC):

- MPEG Surround mode 3 bits
- rfa 2 bits

audio coding field = 01 (reserved) or 10 (reserved):

- rfa 5 bits

MPEG Surround mode: this 3-bit field indicates whether MPEG Surround information is provided along with the AAC core and describes the MPEG Surround target channel setup as follows:

000: no MPEG Surround information available.

001: reserved.

010: MPEG Surround with 5.1 output channels.

011: MPEG Surround with 7.1 output channels.

100: reserved.

101: reserved.

110: reserved.

111: other mode (the mode can be derived from the MPEG Surround data stream).

Receivers with a different number of output channels than the number of target channels indicated by the MPEG Surround mode shall render the multichannel audio signal according to the available number of output channels (possibly at a reduced quality compared to the case where the number of target channels matches the number of output channels).

The number of output channels provided by MPEG Surround is intended as an information for the listener only. The MPEG Surround decoder shall use this information only to determine whether MPEG Surround is used. If MPEG Surround is used (MPEG Surround mode $\neq$ 000), the MPEG Surround decoder shall exclusively rely on the information contained within the SpatialSpecificConfig(). A decoder that does not support MPEG Surround shall ignore this parameter and only decode the mono or stereo core audio.

codec specific config: this field depends upon the value of the audio coding field as follows:

audio coding field = 00 (AAC), 01 (reserved), or 10 (reserved):

- the field is not present ($n = 0$).

audio coding field = 11 (xHE-AAC):

- xHE-AAC Static Config, as returned by the xHE-AAC encoder upon initialization ($n \geq 0$); see clause 5.3.2. The xHE-AAC Static Config bit-field shall be aligned to full bytes by setting the remaining bits of the last byte to 0.

NOTE: A typical value of *n* for mono and stereo configurations without MPEG Surround is in the range 1 to 7 bytes; the inclusion of MPEG Surround adds up to 17 additional bytes.

rfa: these fields are reserved for future additions and shall be set to zero until defined.

6.4.3.11 FAC channel parameters data entity - type 10

This data entity permits the next configuration FAC channel parameters to be specified in advance for service following across reconfigurations. This data entity uses the reconfiguration mechanism for the version flag. The fields are as follows:

- Base/Enhancement flag 1 bit

- Robustness mode 2 bits
- RM flag 1 bit
- Spectrum occupancy 3 bits
- Interleaver depth flag 1 bit
- MSC mode 2 bits
- SDC mode 1 bit
- Number of services 4 bits
- rfa 4 bits
- rfu 1 bit

The following definitions apply:

Base/Enhancement flag: the definition is given in clause 6.3.3.

Robustness mode: this 2-bit field indicates the robustness mode of the new configuration as follows:

RM flag = 0:

00: A.

01: B.

10: C.

11: D.

RM flag = 1:

00: E.

01: reserved.

10: reserved.

11: reserved.

RM flag: the definition is given in clause 6.3.3.

Spectrum occupancy: the definition is given in clause 6.3.3.

Interleaver depth flag: the definition is given in clause 6.3.3.

MSC mode: the definition is given in clause 6.3.3.

SDC mode: the definition is given in clause 6.3.3.

Number of services: the definition is given in clause 6.3.3.

rfa: these 4 bits are reserved for future additions and shall be set to zero until defined.

rfu: this 1 bit is reserved for future use of the whole parameter field and shall be set to zero until defined.

If the DRM transmission is being discontinued at the reconfiguration, then this data entity shall be sent with the length field of the header set to 0, and the first four bits of the body field set to 0.

6.4.3.12 Alternative frequency signalling: Other services data entity - type 11

For every service of the tuned DRM multiplex alternative sources can be signalled. These alternative sources can be DRM services (using different DRM Service identifiers) or services on other broadcast systems like AM, FM, FM-RDS or DAB. For every service of the tuned DRM multiplex, this type of data entity lists the corresponding Service identifier within a DRM multiplex or another broadcast system, optionally along with the list of alternative frequencies. It uses the list mechanism for the version flag. At least one other Service identifier or one frequency shall be provided in this data entity.

NOTE 1: The list of alternative DRM frequencies for the entire DRM multiplex or some of its services (i.e. using the same Service identifiers) can be derived from "Alternative frequency signalling: Multiple frequency network information data entity - type 3".

NOTE 2: For dynamic service linking and delinking of audio services, "Service linking information data entity - type 15, extension 0" can be used.

The alternative frequencies may be scheduled to certain times and/or restricted to certain geographic areas.

The information is as follows:

- Short Id/Announcement Id flag 1 bit
- Short Id/Announcement Id field 2 bits
- Region/Schedule flag 1 bit
- Same Service flag 1 bit
- rfa 2 bits
- System Id 5 bits
- Region/Schedule field 0 bit or 8 bits
- Other Service Id 0 bit or 16 bits or 24 bits or 32 bits
- *n* frequencies $n \times (8 \text{ or } 16)$ bits

The following definitions apply:

Short Id/Announcement Id flag: this flag specifies the content of the Short Id/Announcement Id field as follows:

0: Short Id.

1: Announcement Id.

Short Id/Announcement Id field: the content of this field depends upon the value of the Short Id/Announcement Id flag as follows:

Short Id/Announcement Id flag = 0:

- Short Id 2 bits

Short Id/Announcement Id flag = 1:

- Announcement Id 2 bits

Short Id: this field carries the Short Id of the service in the tuned DRM multiplex to which the alternative frequencies apply.

Announcement Id: this field carries the Announcement Id (value 0 to 3). The information for which services in the tuned DRM multiplex announcements are signalled (and which type of announcements) is described by one or more "Announcement support and switching data entity - type 6" (with the "Same Multiplex/Other Service flag" bit set to 1, indicating that the announcement is provided by another service outside the tuned DRM multiplex).

Region/Schedule flag: this field indicates whether the list of frequencies is restricted by region and/or schedule or not as follows:

0: no restriction.

1: region and/or schedule applies to this list of frequencies.

Same Service flag: this flag indicates whether the specified other service should be considered the "same service" (e.g. carrying the identical audio programme) or an "alternative service" (e.g. a different audio programme either from the same broadcaster offering a similar programme or from another broadcaster - see clauses F.6 and G.1):

0: alternate service.

1: same service.

NOTE 3: When indicating services from another broadcaster; the Short Id field may be set to an arbitrary value. It is recommended that where there are fewer than four services in the multiplex, an unused Short Id should be used.

rfa: these 2 bits are reserved for future additions and shall be set to zero until defined.

System Id: this field indicates which broadcast system the Other Service Id and frequency information applies to as follows:

00000: DRM service
Other Service Id: 24 bits (DRM Service identifier).

00001: AM service with AMSS
Other Service Id: 24 bits (AMSS service identifier).

00010: AM service
Other Service Id: not present (AM service identifier not specified).

00011: FM-RDS service (Europe and North America grid)
Other Service Id: 24 bits (ECC+PI code).

00100: FM-RDS service (Europe and North America grid)
Other Service Id: 16 bits (PI code only).

00101: FM service (Europe and North America grid)
Other Service Id: not present (PI code not specified).

00110: FM-RDS service (Asia grid)
Other Service Id: 24 bits (ECC+PI code).

00111: FM-RDS service (Asia grid)
Other Service Id: 16 bits (PI code only).

01000: FM service (Asia grid)
Other Service Id: not present (PI code not specified).

01001: DAB service
Other Service Id: 24 bits (ECC + audio service identifier).

01010: DAB service
Other Service Id: 16 bits (audio service identifier only).

01011: DAB service
Other Service Id: 32 bits (data service identifier).

all other values: reserved for future definition.

Region/Schedule field: this 8 bit field is only present if the Region/Schedule flag is set to 1. The information is as follows:

- Region Id 4 bits

- Schedule Id 4 bits

Region Id: this field indicates whether the region is unspecified (value 0) or whether the alternative frequencies are valid just in certain geographic areas, in which case it carries the Region Id (value 1 to 15). The region may be described by one or multiple "Alternative frequency signalling: Region definition data entity - type 7" and/or "Alternative frequency signalling: Region definition data entity - type 13" with this Region Id.

Schedule Id: this field indicates whether the schedule is unspecified (value 0) or whether the alternative frequencies are valid just at certain times, in which case it carries the Schedule Id (value 1 to 15). The schedule is described by one or multiple "Alternative frequency signalling: Schedule definition data entity - type 4" with this Schedule Id.

Other Service Id: this field carries the other service identifier used on the *n* frequencies. If an extended country code (ECC) is present, it shall precede the service identifier/PI code. The presence and bit length of this field depends upon the type of broadcast system for which the alternative frequencies are specified (see value of the System Id field for details).

***n* frequencies:** this field carries *n* alternative frequency values (*n* in the range 0 to 16). The bit length and information for every frequency value depends upon the value of the System Id field as follows.

Table 22A

System Id field value	Broadcast system identifier	Frequency value length
00000	DRM frequency	16 bits
00001, 00010	AM frequency	16 bits
00011, 00100, 00101	FM1 frequency	8 bits
00110, 00111, 01000	FM2 frequency	8 bits
01001, 01010, 01011	DAB frequency	8 bits

DRM frequency:

each 16 bit field contains the following information:

- multiplier 1 bit
- frequency value 15 bits

multiplier: this 1-bit field shall indicate the frequency multiplier as follows:

0: 1 (the frequencies field can indicate from 0 to 32 767 kHz in 1 kHz steps; indicated transmission has robustness mode A, B, C or D).

1: 10 (the frequencies field can indicate from 0 to 327 670 kHz in 10 kHz steps; indicated transmission has robustness mode E).

frequency value: this 15 bit field is coded as an unsigned integer and gives the frequency in multiples of 1 or 10 kHz, depending on the value of the multiplier field.

AM frequency:

each 16 bit field contains the following information:

- rfu 1 bit
- frequency value 15 bits

rfu: this 1 bit is reserved for future use of the frequency value field and shall be set to zero until defined.

frequency value: this 15 bit field is coded as an unsigned integer and gives the frequency in kHz.

FM1 (87,5 MHz to 107,9 MHz) frequency:

code	***meaning***
0 to 204:	FM frequencies 87,5 MHz to 107,9 MHz (100 kHz step)

FM2 (76,0 MHz to 90,0 MHz) frequency:

code	*meaning*
0 to 140:	FM frequencies 76,0 MHz to 90,0 MHz (100 kHz step)

DAB [2] frequency:

code	*meaning*	
64 to 95:	DAB channels 5A to 12D	(Band III)
96 to 101:	DAB channels 13A to 13F	(Band III +)

Additional information on Alternative Frequency Signalling is provided in annexes F and G.

6.4.3.13 Language and country data entity - type 12

The language and country data entity allows addition language and country information to be signalled. This data entity uses the unique mechanism for the version flag. The information is as follows:

- Short Id 2 bits
- rfu 2 bits
- language code 24 bits
- country code 16 bits

The following definitions apply:

Short Id: this field indicates the short Id for the service concerned.

rfu: these 2 bits are reserved for future use of the remainder of the parameter field and shall be set to zero until defined.

Language code: this 24-bit field identifies the language of the target audience of the service according to ISO 639-2 [4] using three lower case characters as specified by ISO/IEC 8859-1 [6]. If the language is not specified, the field shall contain three "-" characters.

Country code: this 16-bit field identifies the country of origin of the service (the site of the studio) according to ISO 3166 [5] using two lower case characters as specified by ISO/IEC 8859-1 [6]. If the country code is not specified, the field shall contain two "-" characters.

6.4.3.14 Alternative frequency signalling: detailed region definition data entity - type 13

This data entity allows the definition of geographic areas for which a set of alternative frequencies is provided. This data entity uses the list mechanism for the version flag.

A region can be specified as a geographical area using longitude/latitude and extent values. A range of areas is defined in terms of multiples of $1/16^{th} \times 1/16^{th}$ degree 'squares'. It therefore gives a resolution per square of (EW × NS) 7 km × 7 km (at equator) or 2 km × 7 km at 70° latitude.

This information is as follows:

- Region Id 4 bits
- *m* Squares *m* × 48 bits

Where each Square is defined as:

- rfu 1 bit
- Square Latitude 12 bits

- Square Longitude 13 bits
- Square Latitude Extent 11 bits
- Square Longitude Extent 11 bits

The following definitions apply:

Region Id: this field indicates the identifier for this region definition. Up to 15 different geographic regions with an individual Region Id (values 1 to 15) can be defined; the value 0 shall not be used, since it indicates "unspecified geographic area" in data entity type 3 and 11. The Region Id values are commonly used for SDC data entities 7 and 13; area definitions per Region Id can be specified by any combination of SDC entities 7 and 13 (see table 24 for limitations).

***m* Squares:** this field carries *m* definitions of Squares (*m* in the range 1 to 16). Each Square is coded as a 48-bit field with the following elements:

rfu: this 1 bit is reserved for future use of the remainder of the SDC entity parameter field and shall be set to zero until defined.

Square Latitude: this field specifies the southerly point of the area in 1/16th degrees, as a 2's complement number between -90 (south pole) and +90 (north pole).

Square Longitude: this field specifies the westerly point of the area in 1/16th degrees, as a 2's complement number between -180 (west) and +179 15/16th (east).

Square Latitude Extent: this field specifies the size of the area to the north, in 1/16th degree steps; the value of Latitude plus the value of Latitude Extent shall be equal or less than 90.

Square Longitude Extent: this field specifies the size of the area to the east, in 1/16th degree steps; the value of Longitude plus the value of Longitude Extent may be equal to or exceed the value +180 (i.e. wrap into the region of negative longitude values).

To check whether a certain longitude value is inside the specified longitude range, the formula specified for SDC data entity type 7 shall be used.

This data entity can be used in combination with data entity type 7 which defines the geographic regions with lower resolution as well as the mandatory formula in pseudo program code for evaluating squares.

Additional information on Alternative Frequency Signalling is provided in annexes F and G.

6.4.3.15 Packet stream FEC parameters data entity - type 14

MSC streams carrying data service components in packet mode may have additional error protection added to the packet stream (see clause 6.6.5). The FEC parameters for the packet mode stream are described by this data entity.

This data entity uses the reconfiguration mechanism for the version flag. The general form of the entity is as follows:

- Stream Id 2 bits
- rfu 2 bits
- R 8 bits
- C 8 bits
- packet length 8 bits

The following definitions apply:

Stream Id: this field indicates the stream Id of the stream which has additional error protection applied.

rfu: these 2 bits are reserved for future use of the whole data entity and shall be set to zero until defined.

R: this field, coded as an unsigned binary integer in the range 1 to 180, indicates the value of the R parameter (see clause 6.6.5.1).

C: this field, coded as an unsigned binary integer in the range 1 to 239, indicates the value of the C parameter (see clause 6.6.5.1).

packet length: this field, coded as an unsigned integer in the range 1 to 255, indicates the length in bytes of the data field of each packet (the total packet length is three bytes longer as it includes the header and CRC fields).

NOTE: All packets contained in one data stream have the same length (see clause 6.6.4).

6.4.3.16 Extension data entity - type 15

6.4.3.16.0 General

To permit additional data entities to be defined, data entity type 15 is used. The first four bits of the body of the data entity contain the data entity Extension type field.

6.4.3.16.1 Service linking information data entity - type 15, extension 0

Service linking information can be used to link together services carrying exactly the same content (hard link) or carrying related content (soft link). This data entity uses the list mechanism for the version flag.

The information is as follows:

- Extension type == 0000 4 bits
- Id (Identifier) list flag 1 bit
- LA (Linkage Actuator) 1 bit
- S/H (Soft/Hard) 1 bit
- ILS indicator 1 bit
- LSN (Linkage Set Number) 12 bits
- Rfu 1 bit
- IdLQ (Identifier List Qualifier) 2 bits
- Data flag 1 bit
- Number of Ids: 4 bits
- *n* Ids $n \times$ (16 or 24 or 32) bits

The following definitions apply:

Extension type: this 4-bit field shall be set to 0000 to indicate the correct extension for service linking information.

Id (Identifier) list flag: this 1-bit flag shall indicate whether the *n* Ids, together with the preceding byte, is present or not, as follows:

0: Rfu, IdLQ, Data flag, Number of Ids, and *n* Ids absent;

1: Rfu, IdLQ, Data flag, Number of Ids, and *n* Ids present.

LA (Linkage Actuator): this 1-bit flag shall indicate whether the link is active or inactive (potential), as follows:

0: potential future link or de-activated link;

1: active link.

S/H (Soft/Hard): this 1-bit flag shall indicate whether the link is soft or hard, as follows:

0: Soft link (related services);

1: Hard link (services carrying the same content).

ILS indicator: this 1-bit flag shall indicate whether the link affects only one country (national) or several countries (international), as follows:

0: national link;

1: international link (or DRM or AMSS).

Linkage sets which include services broadcast using DRM or AMSS shall always be an international link, even if all linked services are intended for national use.

NOTE: It is possible but not recommended to signal linkage sets that do not include DRM services.

LSN (Linkage Set Number): this 12-bit field represents a number which shall be common to all Services linked together as a set. The use of LSN = "0000 0000 0000" is reserved.

For an international link, the LSN shall be structured according to IEC 62106 [12].

Rfu: this 1-bit field shall be reserved for future use of the IdLQ, the Rfa, the Number of Ids and the Id list fields. The Rfu bit shall be set to zero for the currently specified definition of the associated fields.

IdLQ (Identifier List Qualifier): this 2-bit field shall indicate how the identifiers, contained in the Id list, are qualified, as follows:

b_6 - b_5

0 0 : each Id represents a DAB service (see ETSI EN 300 401 [2]);

0 1 : each Id represents an RDS service (see IEC 62106 [12]);

1 0 : not used;

1 1 : each Id represents a DRM or AMSS Service (see ETSI TS 102 386 [13]).

Data flag: this 1-bit field shall indicate whether each Id corresponds to a DAB data service or not, as follows:

0: each Id does not correspond to a DAB data service;

1: each Id corresponds to a DAB data service.

Number of Ids: this 4-bit field, expressed as an unsigned binary number, shall specify the number of identifiers in the Id list (maximum 15).

***n* Ids:** this field carries *n* service identifiers (*n* in the range 1 to 15). The bit length and information for every Id value depends upon the value of the IdLQ, ILS indicator and Data flag fields as follows:

Table 22B

IdLQ field value	ILS indicator	Data flag	Id field contains	Id field length
00	1	0	8-bit ECC + 16 bit DAB SId	24 bits
00	1	1	32-bit DAB SId	32 bits
01	1	0	8-bit ECC + 16 bit RDS PI code	24 bits
11	1	0	24-bit DRM or AMSS Service identifier	24 bits
00	0	0	16-bit DAB SId	16 bits
00	0	1	32-bit DAB SId	32 bits
01	0	0	16-bit RDS PI code	16 bits
NOTE: The entries with the ILS indicator set to 0 are for providing linkage sets which only include DAB and/or FM-RDS services.				

The operational use of this data entity is identical to that of the DAB FIG 0/6, see ETSI TS 103 176 [14].

6.4.3.16.2 Other data entity type 15 extensions

Other data entity type 15 extensions are reserved for future definition.

6.4.4 Summary of data entity characteristics

Tables 23 and 24 summarize the version flag mechanism, repetition rate and transmission status of each data entity. The standard repetition rate is that all information for that data entity type should be transmitted within one cycle of the entire database. Individual SDC blocks may carry changed information (e.g. time and date) by use of the FAC identity field.

Table 23: Summary of data entity characteristics

Data entity	Name	Version flag mechanism	Repetition rate
0	Multiplex description	reconfiguration	every SDC block
1	Label	unique	every SDC block
2	Conditional Access Parameters	reconfiguration	as required
3	AFS - Multiple frequency network information	list	standard
4	AFS - Schedule definition	list	standard
5	Application information	reconfiguration	as required
6	Announcement support and switching	list	standard
7	AFS - Region definition	list	standard
8	Time and date information	unique	once per minute
9	Audio information	reconfiguration	every SDC block
10	FAC channel parameters	reconfiguration	every SDC block when FAC reconfiguration index is non-zero
11	AFS - Other services	list	standard
12	Language and country	unique	standard
13	AFS - Region definition	list	standard
14	Packet stream FEC parameters	reconfiguration	every SDC block when FEC for packet mode is used
15/0	Service linking	list	standard

Table 23 gives the recommended repetition rate for fast access to services. However, when the SDC capacity (see clause 6.4.2) is low, lower repetition rates are permitted for every data entity.

Table 24: Summary of data entity characteristics

Data entity	Transmission status	Entity occurrence and limits (normal)	Entity occurrence and limits (during reconfiguration)
0	mandatory	one entity	as defined for (normal) for each configuration; two entities in total
1	optional	zero or one entity per service; zero to four entities in total	as defined for (normal)
2	mandatory for each service for which the FAC CA indication flag = 1	zero, one or two entities per audio service; zero or one entity per data service; zero to seven entities in total	as defined for (normal) for each configuration; zero to 14 entities in total
3	optional	zero to 16 frequencies per entity; zero to 64 entities in total	as defined for (normal)
4	optional	zero to 32 entities per Schedule Id; zero to 128 entities in total	as defined for (normal)
5	mandatory for each data service and data application	zero to four entities per audio service; one entity per data service; zero to 13 entities in total	as defined for (normal) for each configuration; zero to 26 entities in total
6	optional	zero to eight entities in total	as defined for (normal)
7	optional	zero to four entities per Region Id; up to 16 CIRAF zones per Region Id; zero to 32 entities in total	as defined for (normal)
8	optional	zero or one entity per minute	as defined for (normal)
9	mandatory for each audio service	one entity per audio service; zero to four entities in total	as defined for (normal) for each configuration; zero to eight entities in total
10	mandatory when FAC reconfiguration index is non-zero	zero	one entity
11	optional	zero to 16 frequencies per entity; zero to 256 entities in total	as defined for (normal)
12	optional	zero or one entity per service; zero to four entities in total	as defined for (normal)
13	optional	zero to four entities per Region Id; zero to 32 entities in total	as defined for (normal)
14	optional	zero or one entity per packet mode data stream; zero to four entities in total	as defined for (normal) for each configuration; zero to eight entities in total
15/0	optional	zero to 16 entities per service; zero to 64 entities in total	as defined for (normal)

Table 24 lists the transmission characteristics of each SDC entity type. The total number of entities is limited in some cases to ensure consistent receiver operation (by defining the maximum memory size needed).

6.4.5 Changing the content of the SDC

The content of the SDC is important for the operation of Alternative Frequency checking and Switching (AFS). For AFS to function, the receiver has to know what the content of the SDC is in advance so that a correlation may be performed. For this purpose, the AFS index is provided in the SDC and the FAC validates the index by use of the Identity field.

On transmissions with no alternative frequencies, the content of the SDC can be fully dynamic and changed at will: no AFS function is required. In this case it is recommended that the AFS index should be set to 0, and the Identity field in the FAC should then indicate the sequence 11, 01, 10, etc. to indicate that the AFS function cannot be performed.

On transmissions with alternative frequencies, the assignment of data entities to SDC blocks should be carefully designed in order that the content of the SDC can be as static as possible thereby permitting use of the AFS function. In this case it is recommended that the AFS index is chosen such that all required information can be sent in one cycle of SDC blocks. If the content is completely static then the Identity field in the FAC indicates the sequence 00, 01, 10, etc. which indicates that the AFS function can be performed at every position, provided the receiver has stored the data for all the SDC blocks in the cycle.

When the Time and date data entity or announcement support and switching data entity is included in the SDC, and alternative frequencies are signalled, then a semi-dynamic use of the SDC is recommended. In this case one or more SDC blocks in the cycle defined by the AFS index are signalled to be invalid by use of the FAC Identity field thereby allowing the content of those blocks to be changed continuously, whilst other SDC blocks are always signalled as valid by use of the FAC Identity field thereby allowing the AFS function to be performed. An example of changing the SDC content and of using the semi-dynamic scheme with the AFS index = 1 is given in annex G.

A change of the AFS index is only allowed at reconfiguration.

6.4.6 Signalling of reconfigurations

6.4.6.0 Introduction

The DRM Multiplex may be reconfigured at transmission super frame boundaries.

Reconfiguration of the DRM multiplex shall be signalled in advance in order to permit receivers to make the best decisions about how to handle the changes. There are two types of reconfiguration: a service reconfiguration, which concerns the reallocation of the data capacity between the services of the MSC, changes in the MSC protection level or changes in the source coding or data applications; and a channel reconfiguration, which concerns changes to the channel parameters or robustness mode.

Both types of reconfiguration are signalled by setting the FAC reconfiguration index to a non-zero value. The index then counts down on each subsequent transmission super frame. The reconfiguration index shall be identical for all transmission frames of a transmission super frame. The final transmission super frame corresponding to the current configuration shall be that in which the reconfiguration index = 1. The new configuration takes effect for the next transmission super frame and in which the reconfiguration index = 0.

All data entity types that use the reconfiguration mechanism for the version flag that are present in the current configuration, and all data entity types that use the reconfiguration mechanism for the version flag that are required in the new configuration, shall be sent during the period when the reconfiguration index is non-zero with the version flag indicating the next configuration. This shall include data entity type 10 that signals the FAC channel parameters for the new configuration.

6.4.6.1 Service reconfigurations

A service reconfiguration is one in which the data capacity of the MSC is reallocated between services. This happens when the number of services in the multiplex is changed or the size of data streams is changed. A service reconfiguration shall also be signalled if any of the content of the data entity types using the reconfiguration mechanism of the version flag changes. The reconfiguration shall be signalled as far in advance as possible in order to provide the greatest chance that the receiver gets all the information necessary for the next configuration. Therefore the reconfiguration index shall first take the value 7. In certain cases the receiver can follow service reconfigurations without interruption to the audio (see annex Q).

When a new service is introduced, and the overall capacity of the MSC is not changed, then the receiver shall follow the currently selected service through the reconfiguration. To facilitate this, the Service identifier and Short Id of all continuing services shall remain the same. The new service shall use a Short Id that is not used in the current configuration. The one exception to this rule is if there are four services in the current configuration and four services in the new configuration. In this case, if the currently selected service is discontinued, then the receiver follows to the new service with the same Short Id if it is of the same type (e.g. both are audio services).

If the currently selected service is discontinued at the reconfiguration, then the receiver may try to find another source of that service on another frequency and/or system by using the information from data entity types 3 and 11.

6.4.6.2 Channel reconfigurations

A channel reconfiguration is one in which one or more of the following FAC channel parameters are altered: spectrum occupancy, interleaver depth, MSC mode; and/or when the robustness mode is changed. In the case of spectrum occupancy or interleaver depth, the receiver is unable to follow the currently selected service without disruption to the audio output. If the MSC mode is changed the receiver can follow the service without audio interruption as explained in annex Q. The reconfiguration should be signalled as far in advance as possible in order to provide the greatest chance that the receiver gets all the information necessary for the next configuration. Ideally the reconfiguration index should first take the value 7, although a lower starting value may be necessary for operational reasons.

If the transmission is discontinued on the tuned frequency, then a reconfiguration shall be signalled with data entity type 10 taking a special value (see clause 6.4.3.11). In this specific case, the other data entity types that use the reconfiguration mechanism for the version flag shall not be signalled.

6.5 Text message application

Text messages can provide a highly valuable additional element to an audio service without consuming much data capacity. The text message is a basic part of DRM and consumes 80 bits/s in robustness modes A, B, C and D and 320 bits/s in robustness mode E. This capacity can be saved if the service provider does not use text messaging.

The text message (when present) shall occupy the last four bytes of the lower protected part of each logical frame carrying an audio stream. The message is divided into a number of segments. The beginning of each segment of the message is indicated by setting all four bytes to the value 0xFF.

The text message may comprise up to 8 segments. Each segment consists of a header, a body and a CRC. The body shall contain 16 bytes of character data unless it is the last segment in which case it may contain less than 16 bytes.

Each segment is further divided into four-byte pieces which are placed into each successive frame. If the length of the last segment is not a multiple of four then the incomplete frame shall be padded with 0x00 bytes.

When no text message is available for insertion all four bytes shall be set to 0x00.

NOTE: Receivers cannot simply ignore frames with all four bytes set to 0x00 since this may comprise all or part of the CRC and padding bytes which finalize a segment.

UTF-8 character encoding is used. Additional guidance regarding the content and display of text messages is given in clause 6.7.

The structure of the segment is as follows:

- Header 16 bits
- Body $n \times 8$ bits
- CRC 16 bits

The Header is made up as follows:

- toggle bit 1 bit
- first flag 1 bit
- last flag 1 bit
- command flag 1 bit
- field 1 4 bits
- field 2 4 bits
- field 3 4 bits

The following definitions apply:

Toggle bit: this bit shall indicate whether the message or command has changed or is repeated. The Toggle bit shall be considered separately for text messages carrying character data (command flag = 0) and commands (command flag = 1).

- **command flag = 0:** The Toggle bit shall be maintained in the same state for all character data segments from the same text message. When a different text message is sent, this bit shall be inverted with respect to its state for the previous text message. If a text message is repeated, then this bit shall maintain its state.

- **command flag = 1:** The Toggle bit shall be maintained in the same state for all command segments from the same command. When a different command is sent, this bit shall be inverted with respect to its state for the previous command. If a command is repeated, then this bit shall maintain its state.

First flag, Last flag: these flags shall be used to identify particular segments which form a succession of segments in a text message or command. The flags are assigned as follows.

Table 24A

First flag	Last flag	The segment is
0	0	an intermediate segment
0	1	the last segment
1	0	the first segment
1	1	the one and only segment

Command flag: this 1-bit flag signals whether Field 1, Field 2, Field 3 and the Body contain text message related information or command related information, as follows:

0: Field 1, Field 2, Field 3 and the Body contain text message related information including character data;

1: Field 1, Field 2, Field 3 and the Body contain command related information**.**

When the **Command flag = 0** (text message related information):

Field 1:

- **Length:** this 4-bit field, expressed as an unsigned binary number, specifies the number of bytes in the body minus 1. It shall normally take the value 15 except in the last segment.

Field 2:

- if **First flag** = "1":
  - this field contains the value "1111".
- if **First flag** = "0":
  - **rfa:** this 1-bit field is reserved for future additions. The bit shall be set to zero until it is defined.
  - **SegNum** (Segment number)**:** this 3-bit field, expressed as an unsigned binary number, specifies the sequence number of the current segment minus 1. (The second segment of a label corresponds to SegNum = 1, the third segment to SegNum = 2, etc.) The value 0 is reserved for future use.

Field 3:

- if **First flag** = "1":
  - this field contains the **text control** field, see clause 6.7.2.
- if **First flag** = "0":
  - **rfa:** this 4-bit field is reserved for future additions. These bits shall be set to zero until defined.

Body: this field shall be coded as a string of character data (maximum 16 bytes). If the last character of a message segment is a multibyte character and not all bytes fit into the body then the character shall continue in the next message segment. The first character starts with the first byte of the first segment.

When **Command flag = 1** (command related information):

Field 1:

- **Command:** this 4-bit field shall contain a command, as follows (all other codes are reserved for future use):
  - b_{11} to $b_{8;}$
  - 0 0 0 1: clear display command - remove the text message characters from the display;
  - 0 0 1 0: DL Plus command.

When the **Command** field **= 0 0 0 1** (clear display command):

- **Field 2:** this 4-bit field shall contain:
  - **Rfa:** reserved for future additions. The bits shall be set to zero until they are defined.
- **Field 3:** this 4-bit field shall contain:
  - **Rfa:** reserved for future additions. The bits shall be set to zero until they are defined.
- **Body:** this n × 8 bit field shall be omitted.

When the **Command** field **= 0 0 1 0** (DL Plus command):

- **Field 2**, **Field 3** and the **Body**, containing the DL Command field, shall be used as specified in ETSI TS 102 980 [15].

Cyclic Redundancy Check (CRC): this 16-bit CRC shall be calculated on the header and the body. It shall use the generator polynomial $G_{16}(x) = x^{16} + x^{12} + x^5 + 1$.

6.6 Packet mode

6.6.0 Introduction

Data services generally consist of either streams of information, in either synchronous or asynchronous form, or files of information. A generalized packet delivery system allows the delivery of asynchronous streams and files for various services in the same data stream and allows the bit rate of the (synchronous) data stream to be shared on a frame-by-frame basis between the various services. The data stream may be provided with additional error control by the addition of forward error correction. Services can be carried by a series of single packets or as a series of data units. A data unit is a series of packets that are considered as one entity with regard to error handling - one received errored packet within a data unit causes the whole data unit to be rejected. This mechanism can be used to transfer files and also to allow simpler synchronization of asynchronous streams. The carriage of data applications is described in ETSI TS 101 968 [7].

The size of a packet mode data logical frame shall be a multiple of the packet size. The maximum length of a data unit is 8 215 bytes.

6.6.1 Packet structure

6.6.1.0 Introduction

The packet is made up as follows:

- header 8 bits
- data field *n* bytes

- CRC 16 bits

The **header** contains information to describe the packet.

The **data field** contains the data intended for a particular service. The length of the data field is indicated by use of data entity 5, see clause 6.4.3.6.

Cyclic Redundancy Check (CRC): this 16-bit CRC shall be calculated on the header and the data field. It shall use the generator polynomial $G_{16}(x) = x^{16} + x^{12} + x^5 + 1$ (see annex D).

6.6.1.1 Header

The header consists of the following fields:

- first flag 1 bit
- last flag 1 bit
- packet Id 2 bits
- Padded Packet Indicator (PPI) 1 bit
- Continuity Index (CI) 3 bits

The following definitions apply:

First flag, Last flag: these flags are used to identify particular packets which form a succession of packets. The flags are assigned as follows:

Table 24B

First flag	Last flag	The packet is
0	0	an intermediate packet
0	1	the last packet of a data unit
1	0	the first packet of a data unit
1	1	the one and only packet of a data unit

Packet Id: this 2-bit field indicates the Packet Id of this packet.

Padded Packet Indicator: this 1-bit flag indicates whether the data field carries padding or not, as follows:

0: no padding is present: all data bytes in the data field are useful;

1: padding is present: the first byte gives the number of useful data bytes in the data field.

Continuity index: this 3-bit field shall increment by one modulo-8 for each packet with this packet Id.

6.6.1.2 Data field

The data field contains the useful data intended for a particular service.

If the Padded Packet Indicator (PPI) field of the header is 0, then all bytes of the data field are useful bytes.

If the PPI is 1 then the first byte indicates the number of useful bytes that follow, and the data field is completed with padding bytes of value 0x00.

Packets with no useful data are permitted if no packet data is available to fill the logical frame. The PPI shall be set to 1 and the first byte of the data field shall be set to 0 to indicate no useful data. The first and last flags shall be set to 1. The continuity index shall be incremented for these empty packets. If less than 4 sub-streams are used within the data stream then an unused packet id shall be used. Empty packets using a packet id of <p> shall not be inserted during the transmission of a DRM data unit using the same packet id <p>.

6.6.2 Asynchronous streams

Asynchronous streams can be used to transport byte-oriented information. Both single packets and data units can be used to transport asynchronous streams.

Applications that use the single packet transport mechanism shall be able to deal with missing data packets. The first and last flags indicate intermediate packets.

Applications that use the data unit transport mechanism can carry a collection of bytes that are related in a data unit and then make use of the error handling of data units for synchronization purposes.

6.6.3 Files

The file may be carried in a data unit.

Applications that use this transport mechanism shall provide a mechanism to identify each object.

The first and last flags are used to indicate the series of packets that make up the data unit. The continuity index is used to determine whether any intermediate packets have been lost.

6.6.4 Choosing the packet length

A data stream for packet mode may contain one or more packets per logical frame, and the packets may belong to one or more services. However, all packets contained in the stream shall have the same length to minimize the propagation of errors. The choice of the packet length depends on various factors, but the following should be taken into account:

- The overhead of signalling the header and CRC is fixed per packet. Therefore the larger the packet, the lower the ratio of overhead to useful data.
- The amount of padding carried in packets is related to the size of the files compared to the packet size or the transit delay requirements for asynchronous streams. Large packets are less efficient at transporting many small objects.

6.6.5 Forward Error Correction (FEC) for packet mode streams

6.6.5.0 Introduction

Forward Error Correction (FEC), in the form of Reed-Solomon (RS) outer error protection and outer interleaving, can be applied to data streams in packet mode in order to further increase the error robustness of DRM data delivery.

The additional error protection is applied in such a way that receivers not equipped with FEC decoders can still recover the data packets carrying data of the data services, albeit with reduced performance. This is accomplished by creating an FEC frame comprising the unaltered packets (i.e. "data packets") plus additional RS data ("FEC packets") calculated over those data packets as illustrated in figure 12.

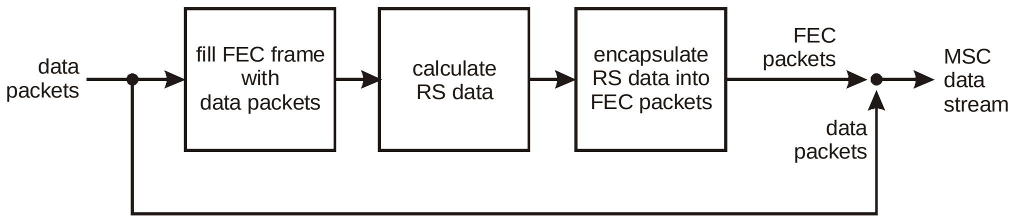


Figure 12: Conceptual diagram of the outer coder and outer interleaver (encoder view)

The input to the functional block is a sequence of data packets from a packet multiplexer. This sequence is referred to as the Application Data Packet Set. All data packets in the Application Data Packet Set have equal packet lengths.

The output of the functional block is the Application Data Packet Set in its original packet order, followed by a number of FEC packets (the FEC Packet Set). The FEC packets contain the RS parity data calculated from the preceding Application Data Packet Set. All FEC packets have the same length as the data packets.

A Packet Set consists of the Application Data Packet Set followed by the FEC Packet Set. A new Packet Set follows immediately after the previous Packet Set.

The FEC scheme protects all data packets in the data stream irrespective of their packet Id value.

A receive terminal applies the reverse process, attempting to correct any transmission errors in the data packets by use of the FEC packets. Packet mode decoders that do not implement the FEC scheme may still recover the data packets for the selected service component by use of the appropriate packet Id and CRC checks and will ignore the FEC packets.

The presence of FEC packets within a packet mode data stream is indicated by SDC data entity type 14 (see clause 6.4.3.15).

6.6.5.1 Encoding of FEC Packets

Figure 13 shows the structure of an FEC frame. This FEC frame has the dimensions of up to 180 rows by up to 255 columns. It consists of an Application Data Table of *R* (1 to 180) rows by *C* (1 to 239) columns plus an RS Data Table of *R* rows by 16 columns. The maximum size of the FEC frame, $M = R \times (C + 16)$, shall not exceed 3 072 bytes to limit receiver cache memory requirements.

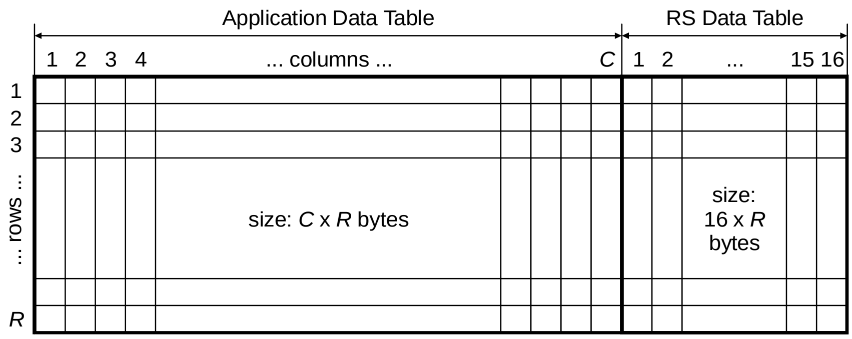


Figure 13: Structure of the FEC frame

The following definitions apply to the values *R* and *C*:

R: the number of rows of the FEC frame, permitted values 1 to 180. Values 0 and 181 to 255 are reserved.

C: the number of columns of the Application Data Table, permitted values 1 to 239. Values 0 and 240 to 255 are reserved.

NOTE 1: The values of *R* and *C* are signalled in SDC data entity type 14.

NOTE 2: The number of columns determines the overhead of the FEC data; the smaller the value of *C* the higher the overhead. The number of rows determines the interleaving depth and the block delay; the smaller the value of *R* the smaller the interleaver and the lower the delay before received data can be processed.

The Application Data Table shall be dimensioned to carry at least one whole packet (i.e. the packet length plus 3 bytes for packet header and CRC value).

The Application Data Table is filled with *D* data packets, where:

$$D = \left\lfloor \frac{R \times C}{L + 3} \right\rfloor$$

and *L* is the packet length provided in SDC data entity 14.

The *D* data packets are consecutively fed into the Application Data Table starting with the first byte of the first packet going into row 1, column 1 and moving downwards row by row, and to the right, column by column. Once all *D* packets have been fed in, any remaining space is filled with padding bytes of value 0x00.

The process is shown in figure 14.

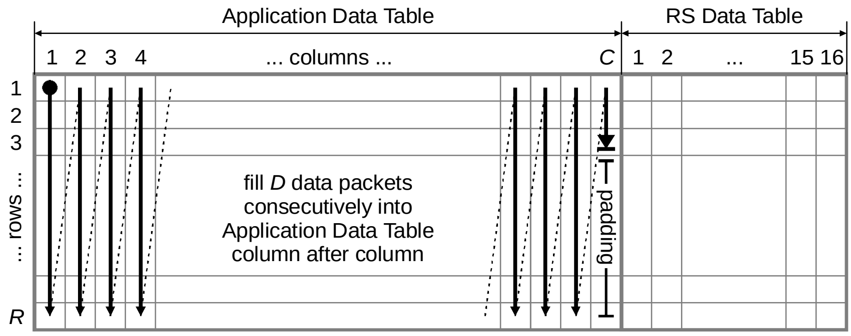


Figure 14: Placing data packets into FEC frame

The RS Data Table is filled by calculating the Reed-Solomon parity data from each row of data from the Application Data Table (i.e. for each Reed-Solomon codeword). The code used for this calculation is the systematic Reed-Solomon RS (255, 239, t = 8) code or a shortened version of this mother code. This code allows the correction of any 8 erroneous bytes anywhere within the codeword.

The following definitions shall apply to calculate the Reed-Solomon parity bytes:

- Code Generator Polynomial: $g(x) = (x+\lambda^0)(x+\lambda^1)(x+\lambda^2)$ to $(x+\lambda^{15})$, where λ = 0x02.
- Field Generator Polynomial: $p(x) = x^8 + x^4 + x^3 + x^2 + 1$.

If the number of columns *C* of the Application Data Table is less than 239, a shortened Reed-Solomon RS ((*C* + 16), *C*, t = 8) shall be used. This shortened Reed-Solomon code may be implemented by adding (239 - *C*) bytes, all set to zero, into the codeword before column 1 of the Application Data Table at the input of an RS (255, 239, t = 8) encoder. After the RS coding procedure these null bytes shall be discarded, leading to a RS codeword length of (*C*+16) bytes.

The process is shown in figure 15.

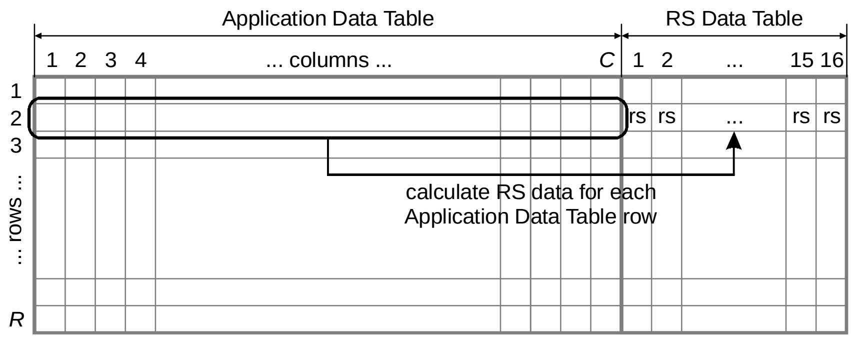


Figure 15: Calculating Reed-Solomon parity data

Finally the Reed-Solomon parity information is encapsulated into FEC packets. The bytes from the RS Data Table are inserted into the FEC packets such that each row of bytes from the RS Data Table is spread over a maximum number of FEC packets.

The FEC packet set consists of *F* FEC packets, where:

$$F = \left\lceil \frac{R \times 16}{L} \right\rceil$$

The RS data bytes of the RS Data Table are transported in the data field of a set of *F* consecutive FEC packets. Each byte of data in the RS Data Table is mapped into successive bytes of the FEC packet data fields, starting with the data byte in row 1, column 1 and working downwards, row by row, and to the right, column by column, until all the data has been mapped (the final byte is from row *R*, column 16 of the RS Data Table). When all the RS data has been mapped, any remaining bytes at the end of the data field (the FEC packet payload) of the last FEC packet shall be filled with zeros.

The procedure is shown in figure 16.

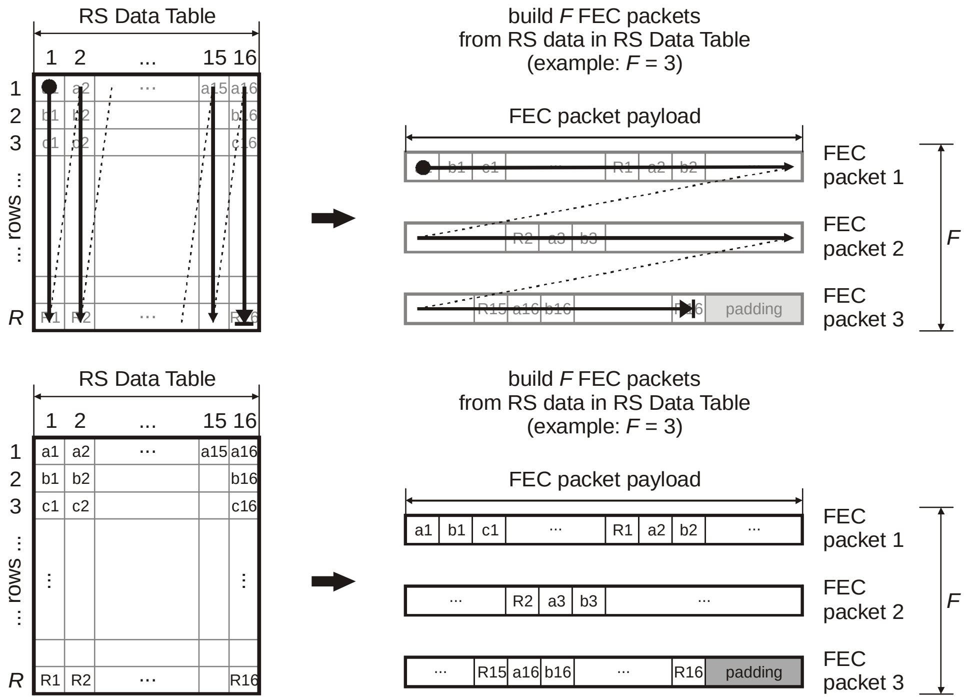


Figure 16: RS data byte order within FEC packets

The following rules apply to FEC packets:

- The first flag shall be set to 1 for the first FEC packet of the FEC Packet Set; otherwise it shall be set to 0.
- The last flag shall be set to 1 for the last FEC packet of the FEC Packet Set; otherwise it shall be set to 0.
- The packet Id shall carry the value 3.
- The Padded Packet Indicator (PPI) shall be set to 0.
- The Continuity Index (CI) shall be set to 0 for the first FEC packet of the FEC Packet Set and then increment for the remaining FEC packets of the FEC Packet Set according to the definition given in clause 6.6.1.1.

NOTE 3: Padding packets as described in clause 6.6.1.2 may be inserted into the packet stream using packet Id 3. These packets are not treated as FEC packets by the FEC encoder and decoder, but as regular data packets. They can be distinguished from FEC packets by their Padded Packet Indicator (PPI) being set to 1 and the first byte of the packet payload carrying the value 0x00.

6.6.5.2 Transport of FEC packets

The set of FEC packets is transmitted immediately following the Application Data Packet Set used to form the Application Data Table. This is visualized in figure 17.

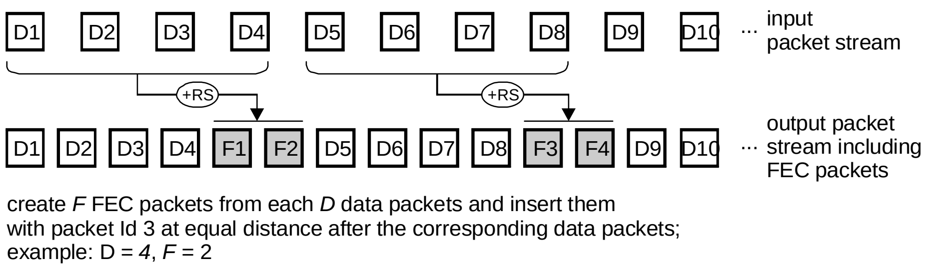


Figure 17: Inserting FEC packets into the packet stream

6.6.5.3 Receiver considerations

The availability of FEC packets in the packet stream is indicated by the presence of SDC data entity type 14 (see clause 6.4.3.15).

The configuration of the FEC scheme is signalled to the decoder by the SDC data entity type 14. This SDC data entity provides the parameters *R* and *C* along with the packet length *L*, so that the FEC decoding can start even before the first SDC data entity type 5 element has been received. Knowing the number of rows *R* and columns *C* of the Application Data Table, the decoder can reconstruct the FEC frame in memory along with the received Reed-Solomon parity bytes by applying the steps to create the FEC packets in reverse order.

The Reed-Solomon error correction mechanism can only take place after all packets belonging to the Application Data Packet Set and corresponding FEC Packet Set have been received. However, if the CRC check passes for a particular data packet, this packet may be used immediately. If the CRC check of a data packet or an FEC packet fails, the receiver may choose to inform the Reed-Solomon decoder of the potentially erroneous byte positions within the Reed-Solomon codeword to enhance the decoder's error correction performance.

The cache memory *M* required in the receiver to collect the received data and FEC packets into an FEC frame is limited to 3 072 bytes.

A receiver can evaluate Reed-Solomon parity information provided as FEC packets after its initial successful synchronization to the packet stream, even if the CRC check for FEC packets fails and therefore the packet headers of these packets cannot be evaluated. This functionality is achieved by inserting the FEC packets with identical FEC configuration and in identical order at equal distances within the packet stream.

Receivers without support for FEC decoding can extract and decode all data packets albeit without the enhanced error correction performance.

6.7 Presentation of service labels and text messages

6.7.1 Introduction

The correct presentation of characters is non-trivial when going beyond basic ASCII: script direction, contextual forms, combining characters and so on make the correct presentation of service labels and text messages a complex task.

Signalling is provided for the label data entity type 1 (see clause 6.4.3.2) and text messages (see clause 6.5) in the form of the **text control** field which provides the base direction of the message and indications of the complexity of the text content. This allows receivers to better determine if they have the necessary capabilities to correctly present the text content.

The use of regional profiles, which include the sets of characters required to support the text content and the presentation capabilities needed by receivers, is supported. Regional profiles should be determined collaboratively by broadcasters, regulators and equipment suppliers - if more capabilities are specified than needed, then receiver products may be overly expensive; if too few capabilities are specified, users will be frustrated. Available regional profiles are given in ETSI TS 103 771 [16].

Most scripts are written and read in a Left-To-Right (LTR) direction, whilst other scripts use a Right-To-Left (RTL) direction; examples are Latin (LTR) and Arabic (RTL). In order to deal with this situation, the Unicode standard [17] defines methods to determine which direction the text should be presented, and rules for when LTR and RTL text are present in the same block of characters, the Unicode bidirectional algorithm, or Bidi for short. Bidi requires that the base direction is specified - that is the direction that should be used when the direction cannot be derived solely from the code points of the text block.

Unicode defines the logical order of the characters as the order in which they are typed. Service labels and text messages are encoded in the Unicode logical order. That means that for LTR only text, the first character will appear left-most in the display, and for RTL only text, the first character will appear right-most on the display. For mixed direction text, the position of the first character of the label is determined by the base direction and the content of the first text block of the label.

EXAMPLE 1: Text (logical order): english words ARABIC WORDS mots français

Base direction	Display shows
LTR	english words SDROW CIBARA mots français
RTL	mots français SDROW CIBARA english words

The first text block is LTR script (shown in lower case), the second is RTL script (shown in UPPER CASE), the third is LTR script: when the base direction is LTR the first text block (LTR) is presented at the left of the display, followed by the second block (RTL), followed by the third block (LTR); when the base direction is RTL the first text block (LTR) is presented at the right of the display, followed by the second block (RTL), followed by the third block (LTR). Each text block is presented in the direction of the script, therefore LTR characters are always presented with the first character of the block on the left, and RTL characters are always presented with the first character of the block on the right.

Numerals, regardless of script, are always presented LTR even when the script direction is RTL. Some characters are designated mirrored characters, such as brackets, quotation marks, etc., whereby a pair of glyphs are mirror images of each other and act as opening and closing marks for a phrase of text. The presentation of these characters is dependent on the script direction.

EXAMPLE 2: Text (logical order): ARABIC 1234 (ABC} WORDS english 1234 [abc» words

Base direction	Display shows
LTR	SDROW {CBA) 1234 CIBARA english 1234 [abc» words
RTL	english 1234 [abc» words SDROW {CBA) 1234 CIBARA

The brackets are mirrored in the RTL text block but not in the LTR text block. The numbers are always presented LTR.

Unicode provides code points for characters but it does not define the glyph that should be used when presenting the character on screen. In many cases the code point can be mapped to a single glyph for a particular font, but in many others, the glyph to be displayed is selected according to the context of the character by analysis of the code points of the surrounding characters. This is particularly true for cursive scripts, such as Arabic, although presentation code point ranges may also be available. These scripts typically have characters that can have markedly different display widths.

Within Unicode, code points exist for base characters and for the same character with particular diacritics. Unicode permits the same character to be represented by following a base character with a combining character.

EXAMPLE 3: e + ˆ= ê (U+0065 followed by U+0302 produces the same glyph as U+00EA).

This type of combining character is called a non-spacing combining character because it does not change the width of the base character. In Indic scripts, there are also spacing combining characters which can extend the width of the character to the left, to the right or to both, and in these scripts the only way to display the script is to create the final glyph by combining glyph parts - presentation forms are not available.

6.7.2 Encoding of the text control field

The **text control** field is made up as follows:

- bidi flag 1 bit
- base direction 1 bit
- contextual flag 1 bit
- combining flag 1 bit

bidi flag: this 1-bit flag shall indicate whether the text contains bidirectional text (excluding numerals) as follows:

0: bidirectional text is not present;

1: bidirectional text is present.

base direction: this 1-bit flag shall define the Unicode base direction of the text as follows:

0: left-to-right (LTR);

1: right-to-left (RTL).

contextual flag: this 1-bit flag shall indicate whether contextual characters are used in the text as follows:

0: contextual characters are not present (presentation characters only);

1: contextual characters are present.

combining flag: this 1-bit flag shall indicate whether combining characters are used in the text as follows:

0: combining characters are not present;

1: combining characters are present.

The **bidi flag** and **base direction** are used to indicate the directional features of the text. They provide information to the receiver about the complexity of the text content.

When the **bidi flag** is set to 0, it indicates that the text is in only one writing direction, either LTR or RTL, as indicated by the **base direction**. It indicates that the text can be presented correctly without the Unicode bidirectional algorithm being implemented. However, RTL labels may include blocks of numeric characters that need to be presented LTR and mirrored characters that need to be presented correctly and so analysis and processing of RTL text is always required. Text with the **bidi flag** set to 0 can also be processed by receivers that implement the Unicode bidirectional algorithm.

When the **bidi flag** is set to 1, it indicates that the text contains both writing directions. To correctly display the text, receivers need to implement either the Unicode bidirectional algorithm or the simplified bidirectional algorithm (see clause 6.7.4.3). In the latter case, only bidirectional text marked explicitly for direction changes using the Left-to-Right-Isolate (LRI) and Right-to-Left-Isolate (RLI) control characters can be displayed correctly.

The **base direction** indicates the default writing direction, regardless of the setting of the **bidi flag**. It shall be set according to the desired writing direction.

The **contextual flag** and **combining flag** are used to indicate whether the text contains characters that change their glyph according to context, and whether the text contains characters that combine to produce a single glyph, respectively. They provide information to the receiver about the complexity of the text content.

When the **contextual flag** is set to 0 it indicates that only presentational code points are present in the label, that is, that there is a 1:1 relationship between a code point and the required glyph (taking account of writing direction for mirrored characters).

When the **contextual flag** is set to 1 it indicates that code points are present in the label that require the displayed glyph to be chosen according to the surrounding characters. This may be according to the character position in the word (for example the isolated, initial, medial or finial forms of Arabic characters) or rendering styles (for example the repeated consonants in south Asian scripts).

When the **combining flag** is set to 0, it indicates no combining characters are present in the text and therefore the receiver is not required to compose the glyph from glyph parts.

When the **combining flag** is set to 1, it indicates that combining characters are present in the text and therefore the receiver is required to be able to compose the glyph from glyph parts. Combining characters can be non-spacing, in which case the width of the base character is unaltered, or spacing, in which case the width of the character may spread to the left, to the right or both. More than one combining character may follow a base character.

6.7.3 Transport of the text control field

6.7.3.1 SDC data entity type 1 labels

For service labels, the **text control** field - when present - is carried as an additional byte at the beginning of the **label** field, see clause 6.4.3.2.

6.7.3.2 Text messages

For text messages, the **text control** field is carried in **Field 3**, see clause 6.5.

6.7.4 Display of RTL and bidirectional text

6.7.4.1 Introduction

Text containing RTL text blocks will have the **base direction** set to RTL and/or the **bidi flag** set to 1. These flags allow a receiver to determine whether it has the capabilities to display the text. Table 24C provides an overview of the required capabilities.

Table 24C: Receiver capability requirements

Bidi flag	Base direction	Label content	Receiver capability needed for presentation
0	0	LTR script(s) only	LTR presentation
0	1	RTL script(s) only	RTL presentation
1	0	LTR script(s) with RTL script block(s)	Bidirectional presentation
1	1	RTL script(s) with LTR script block(s)	Bidirectional presentation

Receivers that implement the Unicode bidirectional algorithm are able to process all labels regardless of the settings of the **bidi flag** and **base direction** (although presentation depends on the settings of the **contextual flag** and **combining flag** and access to glyphs for all contained code points).

Receivers that present RTL scripts shall implement either the Unicode bidirectional algorithm [18] or the minimum RTL implementation (see clause 6.7.4.2) in order to achieve correct presentation. When the Unicode bidirectional algorithm is implemented, it is recommended to enclose the text before display with directional isolates (if implemented, or embedding if not) to take account of the setting of the **base direction**.

Receivers that present bidirectional text (i.e. text with the **bidi flag** set to 1) shall implement either the Unicode bidirectional algorithm [18] or the simplified bidirectional algorithm (see clause 6.7.4.3), the latter requiring the text to contain explicit direction control using directional isolates for proper display.

Receivers that do not implement either the Unicode bidirectional algorithm or the simplified bidirectional algorithm shall not present text with the **bidi flag** set to 1.

Receivers that scroll the display for text message presentation shall take account of the **base direction** and particular care in the presentation of bidirectional text.

6.7.4.2 Minimum RTL implementation

Receivers that do not implement the Unicode bidirectional algorithm may choose to implement the minimum RTL implementation in order to achieve basic support for RTL scripts. This implementation allows text signalled as **bidi flag** = 0, **base direction** = 1 to be correctly formatted (although the settings of the **contextual flag** and **combining flag** may still mean that the label cannot be correctly displayed). In respect of the Unicode bidirectional algorithm [18], when **bidi flag** = 0, **base direction** = 1 the label will not contain any character of bidi class L.

Numeric digits, and their associated characters, are always displayed LTR, even in RTL scripts. The minimum RTL implementation therefore requires numeric digits in the label to be identified, and processing applied in order to present them correctly. Numeric blocks are sequences of characters that include at least one numeric digit, as defined in table 24D. The blocks shall be determined by searching for a numeric digit and then including each character before and after until a space character, U+0020, is found. If the space character has a numeric digit on both sides, then the block shall be extended until the next space character is found, and so on.

Table 24D: Numeric digits

Numeral type	Characters	Unicode codepoints
European digits	0 1 2 3 4 5 6 7 8 9	U+0030 to U+0039
Arabic-Indic digits	٠ ١ ٢ ٣ ٤ ٥ ٦ ٧ ٨ ٩	U+0660 to U+0669
Eastern Arabic-Indic digits	۰ ۱ ۲ ۳ ۴ ۵ ۶ ۷ ۸ ۹	U+06F0 to U+06F9

NOTE 1: This simple algorithm generally produces similar output to the Unicode bidirectional algorithm for text labels when **bidi flag** = 0, **base direction** = 1. However, it cannot correctly deal with all situations.

EXAMPLE: "123", "4 567", "$10.99", "66%", "11/03/2018", "15:20", "14€30", "85°F", "١٢٣,٤٥٦", "۰,۷۸۹" are all numeric blocks.

Some characters are designated mirrored characters whereby a pair of glyphs are mirror images of each other and act as opening and closing marks for a phrase of text. The presentation of these characters is dependent on the writing direction. The minimum RTL implementation therefore requires the identification of the following mirrored characters and for processing to be applied to present them correctly: ( ) [ ] { } < > « ».

NOTE 2: In the Unicode bidirectional algorithm, the *pairs* of mirrored characters are identified as they may be nested and enclose mixed direction scripts. However, since only text with the **bidi flag** set to 0 (or text blocks isolated by the simplified bidirectional algorithm) are processed by the minimum RTL implementation, it is not required to find the pairs since the text is assumed to be well formed. Unicode defines more mirrored characters than those required in the minimum RTL implementation; if these additional characters are present in the text block they will not be mirrored by this implementation.

6.7.4.3 Simplified bidirectional algorithm

Receivers that do not implement the Unicode bidirectional algorithm may choose to implement the simplified bidirectional algorithm in order to achieve basic support for bidirectional text created by service providers according to the rules of the simplified bidirectional algorithm. This implementation allows text signalled as **bidi flag** = 1 to be correctly formatted (although the settings of the **contextual flag** and **combining flag** may still mean that the text cannot be correctly displayed). The simplified bidirectional algorithm requires the implementation of the minimum RTL implementation (see clause 6.7.4.2).

Text encoded using the simplified bidirectional algorithm shall explicitly signal text blocks which are not in the same direction as that indicated by the **base direction**. In this way, receivers are able to detect which parts of the text shall be presented LTR and which shall be presented RTL.

Bidirectional text with the **base direction** indicating LTR shall insert the Right-to-Left-Isolate (RLI, U+2067) control character before an RTL text block and shall insert the Pop-Directional-Isolate (PDI, U+2069) control character at the end of the RTL text block. No other directional control codes shall be used. One or more RTL text blocks may be signalled in a label.

Bidirectional text with the **base direction** indicating RTL shall insert the Left-to-Right-Isolate (LRI, U+2066) control character before the LTR text block and shall insert the Pop-Directional-Isolate (PDI, U+2069) control character at the end of the LTR text block. No other directional control codes shall be used. One or more LTR text blocks may be signalled in a label.

For the examples of bidirectional text provided in clause 6.7.1, the text will be modified as follows when encoded according to the simplified bidirectional algorithm:

EXAMPLE 1: If **base direction** = LTR
Text: english words <RLI>ARABIC WORDS<PDI> mots français
If **base direction** = RTL
Text: <LRI>english words<PDI> ARABIC WORDS <LRI>mots français<PDI>

Base direction	Display shows
LTR	english words SDROW CIBARA mots français
RTL	mots français SDROW CIBARA english words

The position and type of isolates are determined by the base direction of the complete text. The first text block is LTR script (shown in lower case), the second is RTL script (shown in UPPER CASE), the third is LTR script. The control characters are shown as <LRI>, <RLI> and <PDI>. The presentation is identical whether processed by the simple bidirectional algorithm or the Unicode bidirectional algorithm.

EXAMPLE 2: If **base direction** = LTR
Text: <RLI>ARABIC 1234 (ABC} WORDS<PDI> english 1234 [abc» words
If **base direction** = RTL
Text: ARABIC 1234 (ABC} WORDS <LRI>english 1234 [abc» words<PDI>

Base direction	Display shows
LTR	SDROW {CBA) 1234 CIBARA english 1234 [abc» words
RTL	english 1234 [abc» words SDROW {CBA) 1234 CIBARA

The isolates relate to the text blocks and so no isolates are provided for the numeric block within the RTL text block since the minimum RTL algorithm is part of the simplified bidirectional algorithm.

A receiver that implements the simplified bidirectional algorithm shall analyse each label with the **bidi flag** set to 1 and determine whether it contains at least one RLI control code if the **base direction** = LTR or at least one LRI control code if the **base direction** = RTL; if so, the text can be processed, if no such control code is found, then the text cannot be correctly presented since it requires the full Unicode bidirectional algorithm.

When the **base direction** is LTR, the text will be presented from left to right on the receiver display. Providing the first character is not an RLI control code, the first text block is LTR and shall be displayed LTR until the RLI control code is reached. The RLI, at whatever position in the text it occurs, indicates the start of an RTL text block, and the receiver shall determine how much horizontal space is required for this RTL text block by searching for the PDI control code, and use the minimum RTL implementation to correctly deal with numerical blocks and mirrored characters. Characters following the PDI, if any, are LTR and the process repeats until the entire text is displayed.

When the **base direction** is RTL, the text will be presented from right to left on the receiver display. Providing the first character is not an LRI control code, the first text block is RTL and shall be displayed RTL until the LRI control code is reached, using the minimum RTL implementation to correctly deal with numerical blocks and mirrored characters. The LRI, at whatever position in the text it occurs, indicates the start of an LTR text block, and the receiver shall determine how much horizontal space is required for this LTR text block by searching for the PDI control code. Characters following the PDI, if any, are RTL and the process repeats until the entire text is displayed.

NOTE: Receivers that implement the Unicode bidirectional algorithm will correctly process text encoded using the simple bidirectional algorithm and so do not need to follow the simplified bidirectional algorithm search process.

7 Channel coding and modulation

7.1 Introduction

The DRM system consists of three different channels, the MSC, SDC and FAC. Because of the different needs of these channels different coding and mapping schemes shall be applied. An overview of the encoding process is shown in figure 18.

The coding is based on a multilevel coding scheme for which the principle is explained in clause 7.3. Due to different error protection needs within one service or for different services within one multiplex different mapping schemes and combinations of code rates are applicable: Unequal Error Protection (UEP) and Equal Error Protection (EEP) are available. EEP uses a single code rate to protect all the data in a channel. EEP is mandatory for the FAC and SDC. In the MSC, both EEP and UEP can be used, the latter provides two code rates to allow the data to be assigned to a higher protected part and a lower protected part. The application of the coding to the different channels is described in clause 7.5.

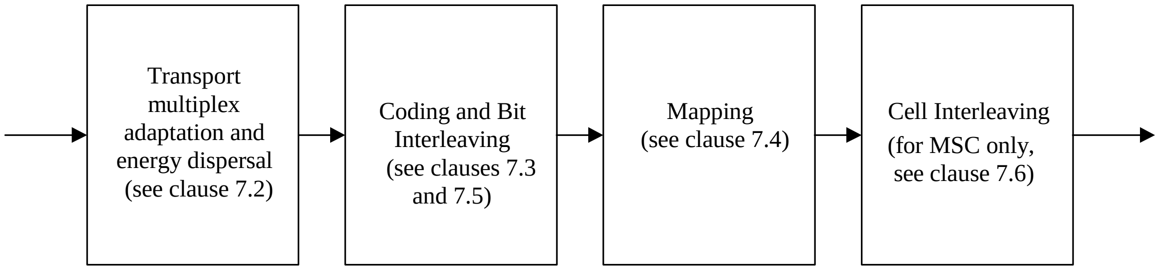


Figure 18: Functional block diagram of the coding and interleaving

7.2 Transport multiplex adaptation and energy dispersal

7.2.1 Transport multiplex adaptation

7.2.1.0 General

The different channels (MSC, SDC, FAC) are processed in the channel coding independently. The vector length L for processing equals one FAC block for the FAC, one SDC block for the SDC or one multiplex frame for the MSC.

7.2.1.1 MSC

The number of bits L_{MUX} per multiplex frame is dependent on the robustness mode, spectrum occupancy and constellation:

- when using one protection level (EEP) it is given by:

$$L_{MUX} = L_2$$

- when using two protection levels (UEP) it is given by:

$$L_{MUX} = L_1 + L_2$$

where the number of bits of the higher protected part is L_1 and the number of bits of the lower protected part is L_2.

L_1 and L_2 are calculated as follows:

$$L_1 = \sum_{p=0}^{P_{\max}-1} 2N_1 R_p$$

$$L_2 = \sum_{p=0}^{P_{\max}-1} RX_p \left\lfloor \frac{2N_2 - 12}{RY_p} \right\rfloor$$

P_{max} is the number of levels (4-QAM: $P_{max} = 1$; 16-QAM: $P_{max} = 2$; 64-QAM: $P_{max} = 3$).

RX_p is the numerator of the code rate of each individual level, see table 27.

RY_p is the denominator of the code rate of each individual level, see table 27.

R_p is the code rate of each individual level, see table 27.

The total number N_{MUX} of MSC OFDM cells per multiplex frame is given in clause 7.7.

The total number N_{MUX} of MSC OFDM cells per multiplex frame when using one protection level (EEP) equals N_2.

The total number N_{MUX} of MSC OFDM cells per multiplex frame when using two protection levels (UEP) equals the addition of the cells of the higher protected part and the lower protected part:

$$N_{MUX} = N_1 + N_2$$

N_1 is the number of OFDM cells used for the higher protected part.

N_2 is the number of OFDM cells used for the lower protected part including the tailbits.

To calculate the number N_1 of OFDM cells in the higher protected part (part A) the following formulae apply:

$$N_1 = \left\lceil \frac{8X}{2RY_{lcm} \sum_{p=0}^{P_{\max}-1} R_p} \right\rceil RY_{lcm}$$

- X is the number of bytes in part A (as signalled in the SDC);
- RY_{lcm} is taken from tables 29 to 32.

$\lceil\ \rceil$ means round towards plus infinity.

To calculate the number N_2 of OFDM cells in the lower protected part (part B) the following formula applies:

$$N_2 = N_{MUX} - N_1$$

The following restrictions shall be taken into account:

$$N_1 \in \{0, \ldots N_{MUX} - 20\}$$

$$N_2 \in \{20, \ldots N_{MUX}\}$$

7.2.1.2 FAC

The number of bits L_{FAC} per FAC block equals 72 bits in robustness modes A, B, C and D and 116 bits in robustness mode E.

The total number N_{FAC} of FAC OFDM cells per FAC block equals 65 in robustness modes A, B, C and D and 244 in robustness mode E.

7.2.1.3 SDC

The number of bits L_{SDC} per SDC block is dependent on the robustness mode, spectrum occupancy and constellation.

The total number N_{SDC} of SDC OFDM cells per SDC block are given in table 25.

The formulas given in clause 7.2.1.1 for the MSC are valid also for the SDC under the constraint of EEP (only 4-QAM: $P_{max} = 1$, 16-QAM: $P_{max} = 2$), i.e. $L_{SDC} = L_2$ and $N_{SDC} = N_2$.

Table 25: Number of QAM cells N_{SDC} for SDC

Robustness mode	Spectrum occupancy					
	0	**1**	**2**	**3**	**4**	**5**
A	167	190	359	405	754	846
B	130	150	282	322	588	662
C	-	-	-	288	-	607
D	-	-	-	152	-	332
E	936	-	-	-	-	-

7.2.2 Energy dispersal

The purpose of the energy dispersal is to avoid the transmission of signal patterns which might result in an unwanted regularity in the transmitted signal.

For the SDC and FAC, the output of the energy dispersal shall form the input stream u_i to the corresponding multilevel coding process.

The output of the energy dispersal acting on the MSC multiplex frame shall form the input stream u_i to the multilevel coding process for the MSC.

Energy dispersal shall be applied on the different channels (MSC, SDC, FAC) in order to reduce the possibility that systematic patterns result in unwanted regularity in either the transmitted signal or in any digital processing, this by providing a deterministic selective complementing of bits.

The individual inputs of the energy dispersal scramblers shown in figure 19 shall be scrambled by a modulo-2 addition with a Pseudo-Random Binary Sequence (PRBS), prior to channel encoding.

The PRBS is defined as the output of the feedback shift register of figure 19. It shall use a polynomial of degree 9, defined by:

$$P(X) = X^9 + X^5 + 1$$

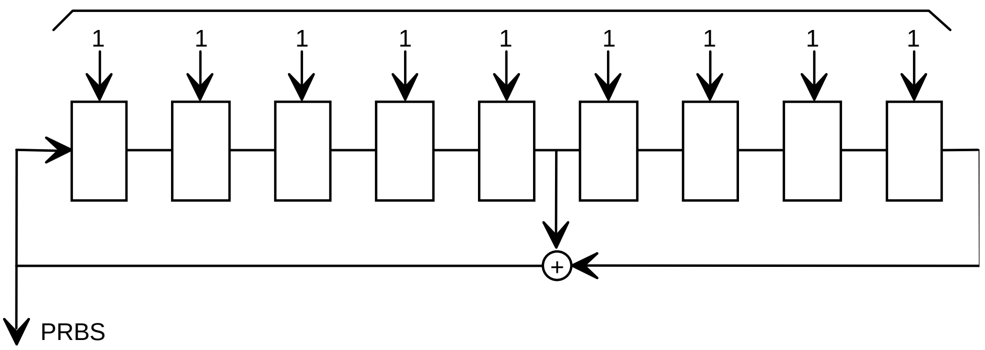


Figure 19: PRBS generator

The initialization word shall be applied in such a way that the first bit of the PRBS is obtained when the outputs of all shift register stages are set to value "1"; the first 16 bits of the PRBS are given in table 26.

Table 26: First 16 bits of the PRBS

bit index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
bit value	0	0	0	0	0	1	1	1	1	0	1	1	1	1	1	0

The FAC, SDC and MSC shall be processed by the energy dispersal scramblers as follows:

- The vector length for processing equals one FAC block for the FAC, one SDC block for the SDC and one multiplex frame for the MSC.

- The block length of the FAC is dependent on the robustness mode; the block lengths for the SDC and MSC are dependent on the robustness mode, spectrum occupancy and constellation, see clause 7.2.1.
- The four blocks shall be processed independently. The input vector shall be scrambled with the PRBS, the first bit of the vector being added modulo 2 to the PRBS bit of index 0.

The scramblers of the different channels are reset as follows:

- FAC: every FAC block;
- SDC: every SDC block;
- MSC: every multiplex frame.

7.3 Coding

7.3.1 Multilevel coding

7.3.1.0 Introduction

The channel encoding process is based on a multilevel coding scheme. The principle of multilevel coding is the joint optimization of coding and modulation to reach the best transmission performance. This denotes that more error prone bit positions in the QAM mapping get a higher protection. The different levels of protection are reached with different component codes which are realized with punctured convolutional codes, derived from the same mother code.

The decoding in the receiver can be done either straightforwardly or through an iterative process. Consequently the performance of the decoder with errored data can be increased with the number of iterations and hence is dependent on the decoder implementation.

Depending on the signal constellation, three different schemes are applicable. The 1-level scheme shall be considered as a special case of the multilevel coding scheme.

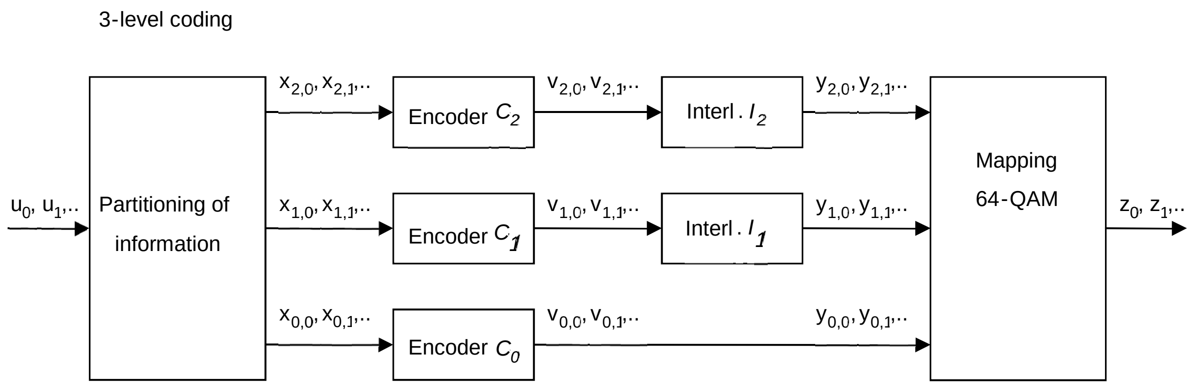


Figure 20: Multilevel coding with 3 levels

Figure 21: Void

Figure 22: Void

2-level coding

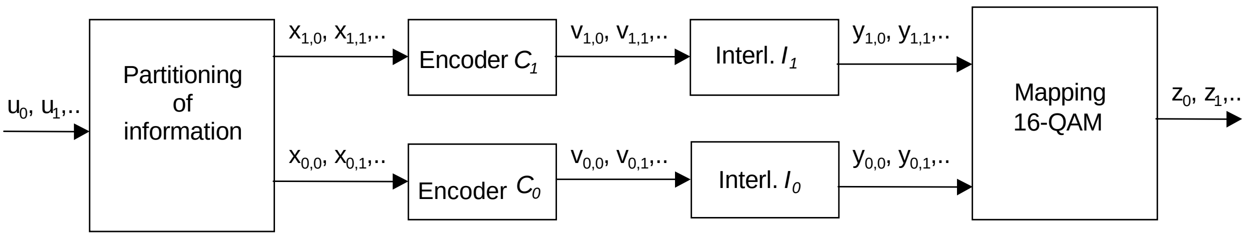


Figure 23: Multilevel coding with 2 levels

1-level coding

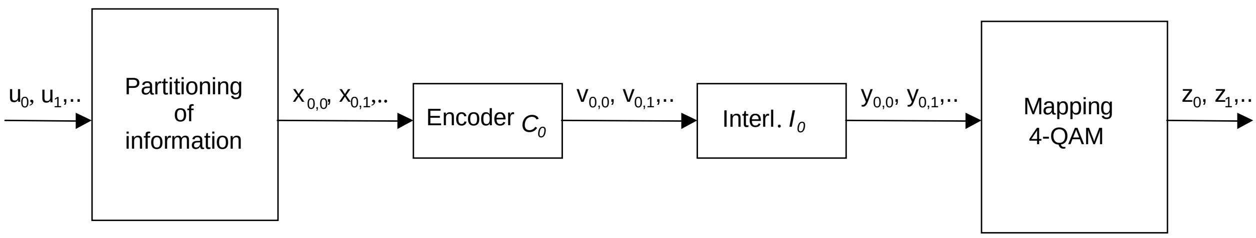


Figure 24: Multilevel coding with 1 level

7.3.1.1 Partitioning of the bitstream

The bitstream (u_i) shall be partitioned into several streams ($x_{p,i}$) according the number of levels. The bits of the higher protected part shall be fed to the encoders on p = 0, to P_{max}-1, then the bits of the lower protected part shall be fed to the encoders on p = 0, to P_{max}-1. This results in:

$$\begin{aligned}&(x_{0,0}, x_{0,1}, \dots x_{0,M_{0,1}-1}, x_{1,0}, x_{1,1}, \dots x_{1,M_{1,1}-1}, x_{2,0}, x_{2,1}, \dots x_{2,M_{2,1}-1}, x_{0,M_{0,1}} x_{0,M_{0,1}+1}, \dots x_{0,M_{0,1}+M_{0,2}-1},\\ &x_{1,M_{1,1}} x_{1,M_{1,1}+1}, \dots x_{1,M_{1,1}+M_{1,2}-1}, x_{2,M_{2,1}} x_{2,M_{2,1}+1}, \dots x_{2,M_{2,1}+M_{2,2}-1}) = (u_0, u_1, \dots u_{L_{VSPP}+L_1+L_2-1})\end{aligned}$$

for the 3-level coding,

$$\begin{aligned}&(x_{0,0}, x_{0,1}, \dots x_{0,M_{0,1}-1}, x_{1,0}, x_{1,1}, \dots x_{1,M_{1,1}-1},\\ &x_{0,M_{0,1}} x_{0,M_{0,1}+1}, \dots x_{0,M_{0,1}+M_{0,2}-1}, x_{1,M_{1,1}} x_{1,M_{1,1}+1}, \dots x_{1,M_{1,1}+M_{1,2}-1}) = (u_0, u_1, \dots u_{L_1+L_2-1})\end{aligned}$$

for the 2-level coding,

$$(x_{0,0}, x_{0,1}, \dots x_{0,M_{0,1}-1}, x_{0,M_{0,1}} x_{0,M_{0,1}+1}, \dots x_{0,M_{0,1}+M_{0,2}-1}) = (u_0, u_1, \dots u_{L_1+L_2-1})$$

for the 1-level coding.

When using only one protection level (EEP) the elements with negative indexes shall not be taken into account.

The number of bits on each level p is calculated for the higher protected part and lower protected part by:

$$M_{p,1} = 2N_1 R_p \text{ where } p \in \{0,1,2\}$$

$$M_{p,2} = RX_p \left\lfloor \frac{2N_2 - 12}{RY_p} \right\rfloor \text{ where } p \in \{0,1,2\}$$

NOTE: The actual number of bits in the higher protected part (L_1) can be greater than the number signalled in the SDC. This means that some bits belonging to part B of the multiplex frame are in fact protected at the higher level.

The total number of bits on each level *p* is:

$$M_p = M_{p,1} + M_{p,2}$$

From these formulas it can be derived that the input bitstreams ($x_{p,i}$) to the encoders C_p have different lengths according to their code rate so that all the encoder output bitstreams ($v_{p,i}$) have the same length.

The overall code rate for each protection part is approximately:

$$R_{all} = \frac{\sum_{p=0}^{P_{\max}-1} R_p}{P_{\max}},$$

when using P_{max} levels.

7.3.2 Component code

The component code C_p is based on punctured convolutional coding with a mother code of rate 1/6 and constraint length 7. The mother convolutional encoder generates from the vector $\left(x_{p,i}\right)_{i=0}^{M_p-1} = \left(a_i\right)_{i=0}^{I-1}$ a codeword $\left\{\left(b_{0,i}, b_{1,i}, b_{2,i}, b_{3,i}, b_{4,i}, b_{5,i}\right)\right\}_{i=0}^{I+5}$. This codeword is defined by:

$$\begin{aligned}
b_{0,i} &= a_i \oplus a_{i-2} \oplus a_{i-3} \oplus a_{i-5} \oplus a_{i-6};\\
b_{1,i} &= a_i \oplus a_{i-1} \oplus a_{i-2} \oplus a_{i-3} \oplus a_{i-6};\\
b_{2,i} &= a_i \oplus a_{i-1} \oplus a_{i-4} \oplus a_{i-6};\\
b_{3,i} &= a_i \oplus a_{i-2} \oplus a_{i-3} \oplus a_{i-5} \oplus a_{i-6};\\
b_{4,i} &= a_i \oplus a_{i-1} \oplus a_{i-2} \oplus a_{i-3} \oplus a_{i-6};\\
b_{5,i} &= a_i \oplus a_{i-1} \oplus a_{i-4} \oplus a_{i-6};
\end{aligned}$$

for i = 0, 1, 2, to I + 5.

When i does not belong to the set {0, 1, 2, to I-1}, a_i is equal to zero by definition.

The encoding can be achieved using the convolutional encoder presented in figure 25.

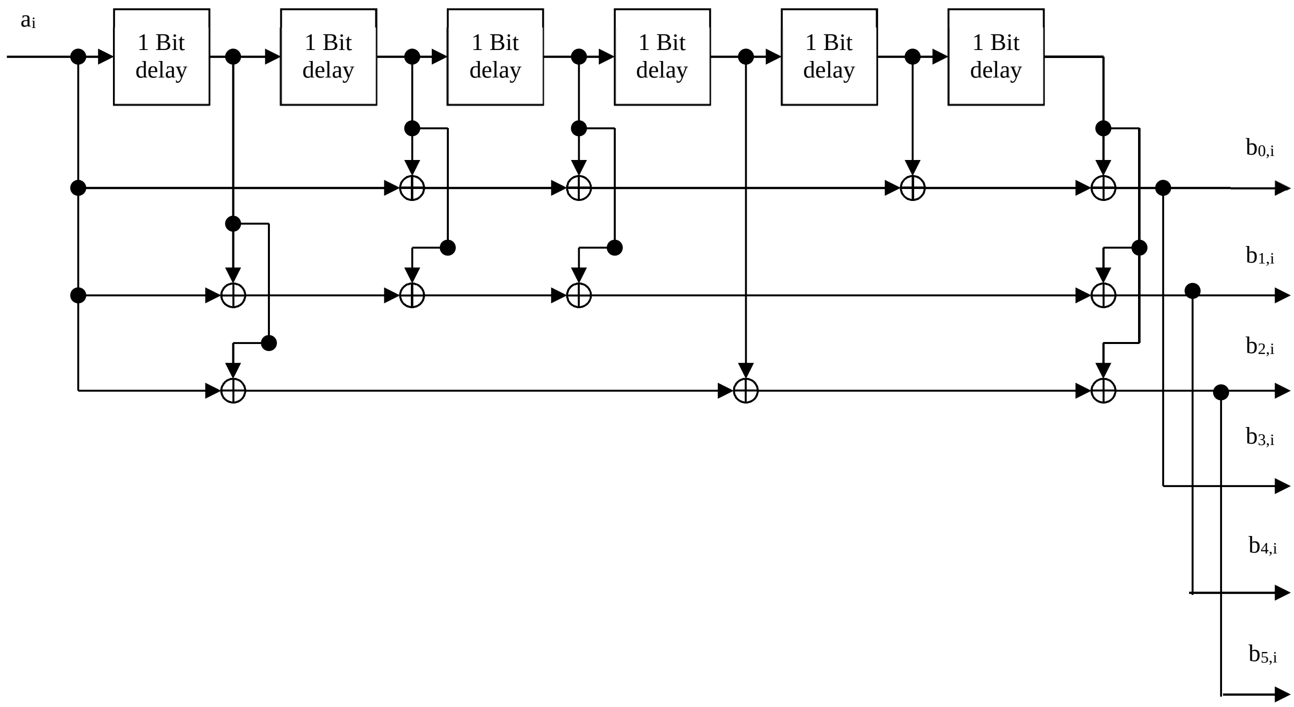


Figure 25: Convolutional encoder

The octal forms of the generator polynomials are 133, 171, 145, 133, 171 and 145, respectively.

The vector $(a_{-6}, a_{-5}, a_{-4}, a_{-3}, a_{-2}, a_{-1})$ corresponds to the all-zero initial state of the shift register and the vector $(a_I, a_{I+1}, a_{I+2}, a_{I+3}, a_{I+4}, a_{I+5})$ corresponds to the all-zero final state of the shift register.

In addition to the mother code the system shall allow punctured rates. Table 27 shows the puncturing patterns.

Table 27: Puncturing patterns

Code rates R_p	Numerator RX_p	Denominator RY_p	Puncturing pattern	Transmitted sequence
1/6	1	6	B_0: 1 B_1: 1 B_2: 1 B_3: 1 B_4: 1 B_5: 1	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{3,0}$ $b_{4,0}$ $b_{5,0}$ etc.
1/4	1	4	B_0: 1 B_1: 1 B_2: 1 B_3: 1 B_4: 0 B_5: 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{3,0}$ etc.
3/10	3	10	B_0: 1 1 1 B_1: 1 1 1 B_2: 1 1 1 B_3: 1 0 0 B_4: 0 0 0 B_5: 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{3,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{0,2}$ $b_{1,2}$ $b_{2,2}$ etc.

Code rates R_p	Numerator RX_p	Denominator RY_p	Puncturing pattern	Transmitted sequence
1/3	1	3	B_0: 1 B_1: 1 B_2: 1 B_3: 0 B_4: 0 B_5: 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ etc.
4/11	4	11	B_0: 1 1 1 1 B_1: 1 1 1 1 B_2: 1 1 1 0 B_3: 0 0 0 0 B_4: 0 0 0 0 B_5: 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{0,2}$ $b_{1,2}$ $b_{2,2}$ $b_{0,3}$ $b_{1,3}$ etc.
2/5	2	5	B_0: 1 1 B_1: 1 1 B_2: 1 0 B_3: 0 0 B_4: 0 0 B_5: 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{0,1}$ $b_{1,1}$ etc.
1/2	1	2	B_0: 1 B_1: 1 B_2: 0 B_3: 0 B_4: 0 B_5: 0	$b_{0,0}$ $b_{1,0}$ etc.
4/7	4	7	B_0: 1 1 1 1 B_1: 1 0 1 0 B_2: 0 1 0 0 B_3: 0 0 0 0 B_4: 0 0 0 0 B_5: 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{0,1}$ $b_{2,1}$ $b_{0,2}$ $b_{1,2}$ $b_{0,3}$ etc.
3/5	3	5	B_0: 1 1 1 B_1: 1 0 1 B_2: 0 0 0 B_3: 0 0 0 B_4: 0 0 0 B_5: 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{0,1}$ $b_{0,2}$ $b_{1,2}$ etc.
2/3	2	3	B_0: 1 1 B_1: 1 0 B_2: 0 0 B_3: 0 0 B_4: 0 0 B_5: 0 0	$b_{0,0}$ $b_{1,0}$ $b_{0,1}$, etc.
8/11	8	11	B_0: 1 1 1 1 1 1 1 1 B_1: 1 0 0 1 0 0 1 0 B_2: 0 0 0 0 0 0 0 0 B_3: 0 0 0 0 0 0 0 0 B_4: 0 0 0 0 0 0 0 0 B_5: 0 0 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$ $b_{1,3}$ $b_{0,4}$ $b_{0,5}$ $b_{0,6}$ $b_{1,6}$ $b_{0,7}$, etc.

Code rates R_p	Numerator RX_p	Denominator RY_p	Puncturing pattern	Transmitted sequence
3/4	3	4	B_0: 1 1 1 B_1: 1 0 0 B_2: 0 0 0 B_3: 0 0 0 B_4: 0 0 0 B_5: 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{0,1}$ $b_{0,2}$, etc.
4/5	4	5	B_0: 1 1 1 1 B_1: 1 0 0 0 B_2: 0 0 0 0 B_3: 0 0 0 0 B_4: 0 0 0 0 B_5: 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$ $b_{0,4}$, etc.
7/8	7	8	B_0: 1 1 1 1 1 1 1 B_1: 1 0 0 0 0 0 0 B_2: 0 0 0 0 0 0 0 B_3: 0 0 0 0 0 0 0 B_4: 0 0 0 0 0 0 0 B_5: 0 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$ $b_{0,4}$ $b_{0,5}$ $b_{0,6}$, etc.
8/9	8	9	B_0: 1 1 1 1 1 1 1 1 B_1: 1 0 0 0 0 0 0 0 B_2: 0 0 0 0 0 0 0 0 B_3: 0 0 0 0 0 0 0 0 B_4: 0 0 0 0 0 0 0 0 B_5: 0 0 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$ $b_{0,4}$ $b_{0,5}$ $b_{0,6}$ $b_{0,7}$, etc.

For the FAC, all bits are punctured according to table 27. For the MSC and the SDC, the last 36 bits (the tailbits) of the serial mother codeword shall be punctured as follows. The index r_p shall be used with table 27 to find the puncturing vector for the tailbits for each level. This index is calculated with the following formula:

$$r_p = (2N_2 - 12) - RY_p \left\lfloor \frac{2N_2 - 12}{RY_p} \right\rfloor \text{ for } p \in \{0,1,2\}$$

Table 28: Puncturing patterns of the tailbits

r_p	Puncturing pattern	Transmitted sequence
0	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 0 0 0 0 0 0 B_3: 0 0 0 0 0 0 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{0,1}$ $b_{1,1}$ $b_{0,2}$ $b_{1,2}$ $b_{0,3}$ $b_{1,3}$ $b_{0,4}$ $b_{1,4}$ $b_{0,5}$ $b_{1,5}$, etc.
1	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 0 0 0 0 0 B_3: 0 0 0 0 0 0 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{0,1}$ $b_{1,1}$ $b_{0,2}$ $b_{1,2}$ $b_{0,3}$ $b_{1,3}$ $b_{0,4}$ $b_{1,4}$ $b_{0,5}$ $b_{1,5}$, etc.

r_p	Puncturing pattern	Transmitted sequence
2	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 0 0 1 0 0 B_3: 0 0 0 0 0 0 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{0,1}$ $b_{1,1}$ $b_{0,2}$ $b_{1,2}$ $b_{0,3}$ $b_{1,3}$ $b_{2,3}$ $b_{0,4}$ $b_{1,4}$ $b_{0,5}$ $b_{1,5}$, etc.
3	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 1 0 1 0 0 B_3: 0 0 0 0 0 0 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{0,2}$ $b_{1,2}$ $b_{0,3}$ $b_{1,3}$ $b_{2,3}$ $b_{0,4}$ $b_{1,4}$ $b_{0,5}$ $b_{1,5}$, etc.
4	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 1 0 1 1 0 B_3: 0 0 0 0 0 0 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{0,2}$ $b_{1,2}$ $b_{0,3}$ $b_{1,3}$ $b_{2,3}$ $b_{0,4}$ $b_{1,4}$ $b_{2,4}$ $b_{0,5}$ $b_{1,5}$, etc.
5	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 1 1 1 1 0 B_3: 0 0 0 0 0 0 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{0,2}$ $b_{1,2}$ $b_{2,2}$ $b_{0,3}$ $b_{1,3}$ $b_{2,3}$ $b_{0,4}$ $b_{1,4}$ $b_{2,4}$ $b_{0,5}$ $b_{1,5}$, etc.
6	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 1 1 1 1 1 B_3: 0 0 0 0 0 0 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{0,2}$ $b_{1,2}$ $b_{2,2}$ $b_{0,3}$ $b_{1,3}$ $b_{2,3}$ $b_{0,4}$ $b_{1,4}$ $b_{2,4}$ $b_{0,5}$ $b_{1,5}$ $b_{2,5}$, etc.
7	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 1 1 1 1 1 B_3: 1 0 0 0 0 0 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{3,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{0,2}$ $b_{1,2}$ $b_{2,2}$ $b_{0,3}$ $b_{1,3}$ $b_{2,3}$ $b_{0,4}$ $b_{1,4}$ $b_{2,4}$ $b_{0,5}$ $b_{1,5}$ $b_{2,5}$, etc.
8	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 1 1 1 1 1 B_3: 1 0 0 1 0 0 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{3,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{0,2}$ $b_{1,2}$ $b_{2,2}$ $b_{0,3}$ $b_{1,3}$ $b_{2,3}$ $b_{3,3}$ $b_{0,4}$ $b_{1,4}$ $b_{2,4}$ $b_{0,5}$ $b_{1,5}$ $b_{2,5}$, etc.
9	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 1 1 1 1 1 B_3: 1 1 0 1 0 0 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{3,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{3,1}$ $b_{0,2}$ $b_{1,2}$ $b_{2,2}$ $b_{0,3}$ $b_{1,3}$ $b_{2,3}$ $b_{3,3}$ $b_{0,4}$ $b_{1,4}$ $b_{2,4}$ $b_{0,5}$ $b_{1,5}$ $b_{2,5}$, etc.

r_p	Puncturing pattern	Transmitted sequence
10	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 1 1 1 1 1 B_3: 1 1 0 1 0 1 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{3,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{3,1}$ $b_{0,2}$ $b_{1,2}$ $b_{2,2}$ $b_{0,3}$ $b_{1,3}$ $b_{2,3}$ $b_{3,3}$ $b_{0,4}$ $b_{1,4}$ $b_{2,4}$ $b_{0,5}$ $b_{1,5}$ $b_{2,5}$, $b_{3,5}$, etc.
11	B_0: 1 1 1 1 1 1 B_1: 1 1 1 1 1 1 B_2: 1 1 1 1 1 1 B_3: 1 1 1 1 0 1 B_4: 0 0 0 0 0 0 B_5: 0 0 0 0 0 0	$b_{0,0}$ $b_{1,0}$ $b_{2,0}$ $b_{3,0}$ $b_{0,1}$ $b_{1,1}$ $b_{2,1}$ $b_{3,1}$ $b_{0,2}$ $b_{1,2}$ $b_{2,2}$, $b_{3,2}$, $b_{0,3}$ $b_{1,3}$ $b_{2,3}$ $b_{3,3}$ $b_{0,4}$ $b_{1,4}$ $b_{2,4}$ $b_{0,5}$ $b_{1,5}$ $b_{2,5}$, $b_{3,5}$, etc.

The puncturing shall be performed as follows:

The higher protected part of the transmitted sequence is punctured according to table 27 resulting in:

$$\left(v_{p,0} \dots v_{p,2N_1-1}\right)$$

The lower protected part of the transmitted sequence is punctured according to table 27 resulting in:

$$\left(v_{p,2N_1} \dots v_{p,2(N_1+N_2)-13-r_p}\right)$$

The tailbits of the transmitted sequence are punctured according to table 28 resulting in:

$$\left(v_{p,2(N_1+N_2)-12-r_p} \dots v_{p,2(N_1+N_2)-1}\right)$$

NOTE: If there is only one protection level the higher protected part is absent.

7.3.3 Bit interleaving

7.3.3.0 Introduction

Bit-wise interleaving shall be applied for some of the levels of the coding scheme according to figures 20 to 24. The same basic algorithm which results in a pseudo random bit ordering shall be used independently for the FAC, SDC and MSC.

The permutation $\Pi_p(i)$ is obtained from the following relations:

$$\text{for 64-QAM: } t_1 = 13,\ t_2 = 21$$

$$\text{for 16-QAM: } t_0 = 13,\ t_1 = 21$$

$$\text{for 4-QAM: } t_0 = 21$$

$$p \in \{0,1,2\}$$

$$s = 2^{\lceil \log 2(x_{in}) \rceil}$$

$$q = s/4 - 1$$

the number of input bits x_{in} is defined below and $\lceil\ \rceil$ means round towards plus infinity.

$\Pi_p(0) = 0$;

for $i = 1, 2, \ldots, x_{in} - 1$:

$$\Pi_p(i) = \left(t_p\ \Pi_p(i-1) + q\right)(\bmod s);$$

while $\Pi_p(i) \geq x_{in}$:

$$\Pi_p(i) = \left(t_p\ \Pi_p(i) + q\right)(\bmod s).$$

7.3.3.1 FAC

The block size shall be in every case the same for the interleaver I_p with p = 0 only. The number of elements per bit interleaver x_{in} equals $2N_{FAC}$. The input vector is defined by:

$$V(p) = \left(v_{p,0}, v_{p,1}, v_{p,2}, \ldots v_{p,2N_{FAC}-1}\right)$$

The interleaved output vector is the subset of the permutations $\Pi_p(i)$ defined by:

$$Y(p) = \left(y_{p,0}, y_{p,1}, y_{p,2}, \ldots y_{p,2N_{FAC}-1}\right)$$

The output elements are selected from the input elements according to:

$$y_{p,i} = v_{p,\Pi_p(i)}$$

7.3.3.2 SDC

The block size shall be the same for each interleaver I_p. The number of elements per bit interleaver x_{in} equals $2N_{SDC}$. For each bit interleaver, the input vector is defined by:

$$V(p) = \left(v_{p,0}, v_{p,1}, v_{p,2}, \ldots v_{p,2N_{SDC}-1}\right)$$

The interleaved output vector is the subset of the permutations $\Pi_p(i)$ defined by:

$$Y(p) = \left(y_{p,0}, y_{p,1}, y_{p,2}, \ldots y_{p,2N_{SDC}-1}\right)$$

The output elements are selected from the input elements according to:

$$y_{p,i} = v_{p,\Pi_p(i)}.$$

7.3.3.3 MSC

The block size shall be the same for each interleaver I_p, but shall be dependent on the robustness mode, spectrum occupancy and the constellation. The number of elements per bit interleaver equals $2(N_1 + N_2)$. For each bit interleaver, the input vector is defined by:

$$V(p) = \left(v_{p,0}, v_{p,1}, v_{p,2}, \ldots v_{p,2(N_1+N_2)-1}\right) = \left(v_{1,p,0}, v_{1,p,1}, \ldots v_{1,p,2N_1-1}, v_{2,p,0}, v_{2,p,1}, \ldots v_{2,p,2N_2-1}\right)$$

The interleaved output vector is the subset of the two permutations $\Pi_p(i)$ defined by:

$$Y(p) = \left(y_{p,0}, y_{p,1}, y_{p,2}, \ldots y_{p,2(N_1+N_2)-1}\right) = \left(y_{1,p,0}, y_{1,p,1}, \ldots y_{1,p,2N_1-1}, y_{2,p,0}, y_{2,p,1}, \ldots y_{2,p,2N_2-1}\right)$$

The two parts with different protection levels shall not overlap in the interleaving process. Therefore the interleaved lower protected part shall be appended to the interleaved higher protected part where the output elements are selected from the input elements according to:

$$y_{1,p,i} = v_{1,p,\Pi_p(i)} \text{ and } y_{2,p,i} = v_{2,p,\Pi_p(i)}$$

for each part respectively.

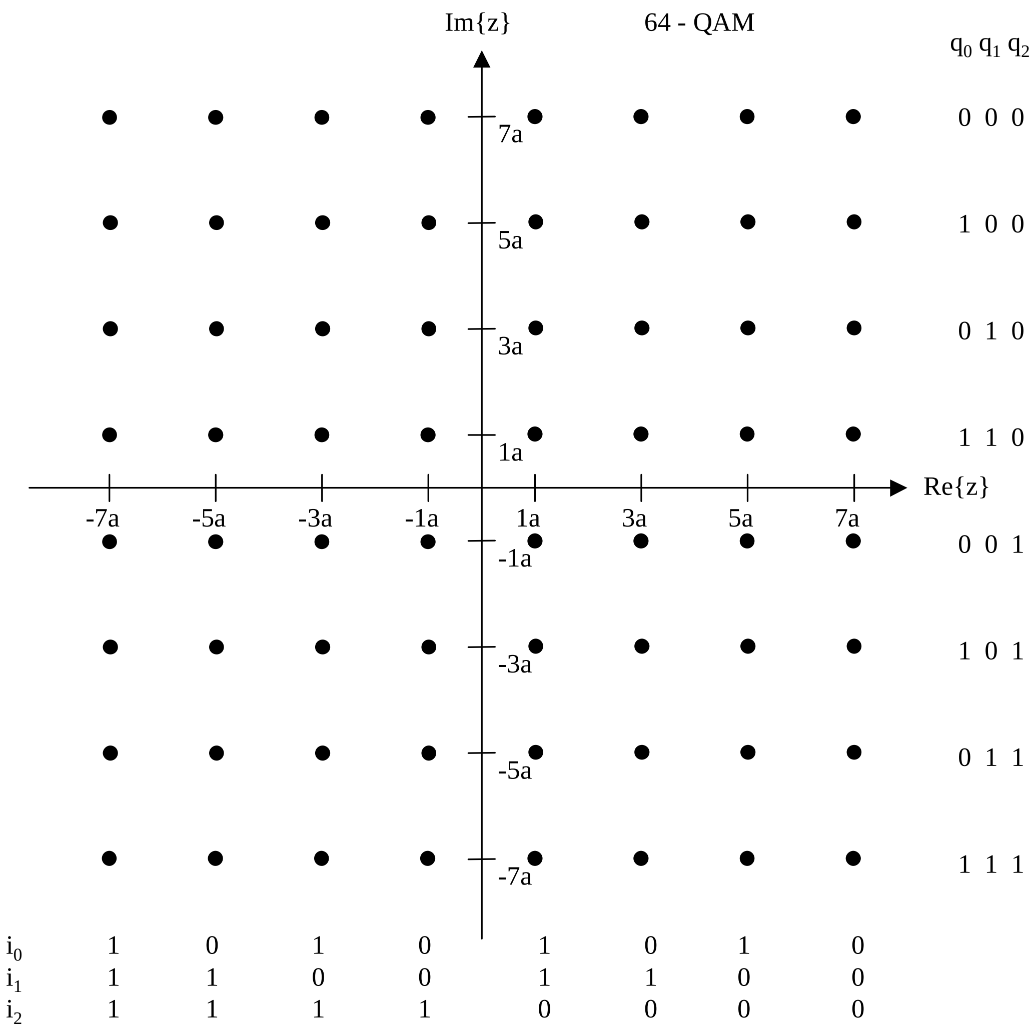
The number of input bits used for the permutation for the higher protected part is $x_{in} = 2N_1$, and for the lower protected part is $x_{in} = 2N_2$.

7.4 Signal constellations and mapping

The mapping strategy for each OFDM cell is dependent of the assignment to the channel (FAC, SDC and MSC) and the robustness mode. All data cells are either 4-QAM, 16-QAM or 64-QAM.

The default method for mapping shall be performed according to figures 26, 29 and 30.

The y'_i denotes the bits representing a complex modulation symbol z.

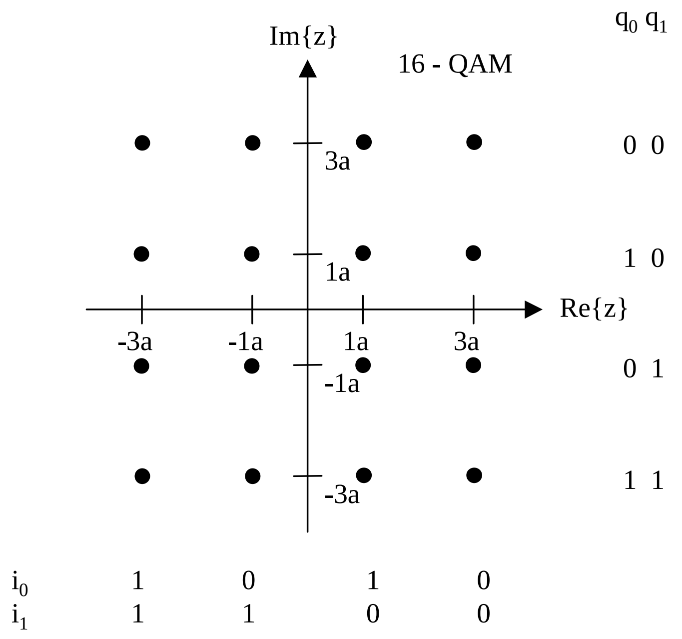


Bit ordering: $\{i_0\ i_1\ i_2\ q_0\ q_1\ q_2\} = \{y'_0\ y'_1\ y'_2\ y'_3\ y'_4\ y'_5\}$

Figure 26: 64-QAM mapping with corresponding bit pattern

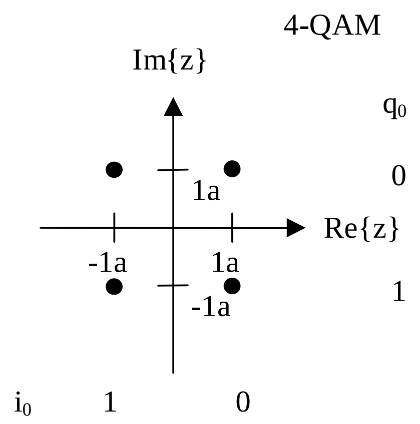
Figure 27: Void

Figure 28: Void



Bit ordering: $\{i_0\, i_1\, q_0\, q_1\} = \{y'_0\, y'_1\, y'_2\, y'_3\}$

Figure 29: 16-QAM mapping with corresponding bit pattern



Bit ordering: $\{i_0\, q_0\} = \{y'_0\, y'_1\}$

Figure 30: 4-QAM mapping with corresponding bit pattern

NOTE: Left hand bit is the first in time.

For 64-QAM, the normalization factor is $a = \frac{1}{\sqrt{42}}$.

For 16-QAM, the normalization factor is $a = \frac{1}{\sqrt{10}}$.

For 4-QAM, the normalization factor is $a = \frac{1}{\sqrt{2}}$.

The data stream at the output of the interleaver consists of a number of bit words. These are mapped onto one signal point in the signal diagram according a complex number z. The 64-QAM diagram shall be used according to figure 26. The bits shall be mapped accordingly:

$$\left(y_0^{'}\ y_1^{'}\ y_2^{'}\ y_3^{'}\ y_4^{'}\ y_5^{'}\right)=\left(y_{0,0}\ y_{1,0}\ y_{2,0}\ y_{0,1}\ y_{1,1}\ y_{2,1}\right)$$

The 16-QAM diagram shall be used according figure 29. The bits shall be mapped accordingly:

$$\left(y_0^{'}\ y_1^{'}\ y_2^{'}\ y_3^{'}\right)=\left(y_{0,0}\ y_{1,0}\ y_{0,1}\ y_{1,1}\right)$$

The 4-QAM diagram shall be used according figure 30. The bits shall be mapped accordingly:

$$\left(y_0^{'}\ y_1^{'}\right)=\left(y_{0,0} y_{0,1}\right)$$

7.5 Application of coding to the channels

7.5.1 Coding the MSC

7.5.1.0 Introduction

The MSC may use either 64-QAM or 16-QAM mapping in robustness modes A, B, C and D and 16-QAM or 4-QAM mapping in robustness mode E. For all robustness modes, the higher constellation provides high spectral efficiency whereas the lower constellation provides a more robust error performance.

In each case, a range of code rates is available to provide the most appropriate level of error correction for a given transmission. The available combinations of constellation and code rate provide a large degree of flexibility over a wide range of transmission channels. Unequal error protection can be used to provide two levels of protection for the MSC.

7.5.1.1 Coding

Two protection levels within one multiplex frame are possible resulting in the use of two overall code rates. The number of input bits L_{MUX} per multiplex frame is calculated with the formulas of clause 7.2.

The MSC shall be encoded according to clause 7.3. The overall code rates and code rates for each level are defined in tables 29 to 32. The protection level is signalled in the multiplex description data entity of the SDC (see clause 6.4.3.1).

Four code rates are defined for 4-QAM for robustness mode E as follows.

Table 29: Code rates for the MSC with 4-QAM (robustness mode E)

Protection level	Rall	R_0
0	0,25	1/4
1	0,33	1/3
2	0,4	2/5
3	0,5	1/2

Two overall code rates are defined for 16-QAM for robustness modes A, B, C and D and four overall code rates are defined for 16-QAM for robustness mode E as follows.

Table 30: Code rate combinations for the MSC with 16-QAM (robustness modes A, B, C and D)

Protection level	Rall	R_0	R_1	RY_{lcm}
0	0,5	1/3	2/3	3
1	0,62	1/2	3/4	4

Table 31: Code rate combinations for the MSC with 16-QAM (robustness mode E)

Protection level	Rall	R_0	R_1	RY_{lcm}
0	0,33	1/6	1/2	6
1	0,41	1/4	4/7	28
2	0,5	1/3	2/3	3
3	0,62	1/2	3/4	4

Four overall code rates are defined for 64-QAM as follows.

Table 32: Code rate combinations for the MSC with 64-QAM (robustness modes A, B, C and D)

Protection level	Rall	R_0	R_1	R_2	RY_{lcm}
0	0,5	1/4	1/2	3/4	4
1	0,6	1/3	2/3	4/5	15
2	0,71	1/2	3/4	7/8	8
3	0,78	2/3	4/5	8/9	45

One or two overall code rates shall be applied to one multiplex frame. When using two overall code rates, both shall belong to the same constellation.

Annex J provides the number of input bits per multiplex frame for EEP.

Table 33: Void

Table 34: Void

Table 35: Void

7.5.2 Coding the SDC

The SDC may use either 16-QAM or 4-QAM mapping with code rate 0,5 for robustness modes A, B, C and D and 4-QAM mapping with code rate 0,5 or 0,25 for robustness mode E. In each robustness mode, a choice is available between greater capacity and a more robust error performance. In each case, a fixed code rate is applied.

The constellation and code rate should be chosen with respect to the MSC parameters to provide more robustness for the SDC than for the MSC.

The number of input bits L_{SDC} per SDC block is calculated as given in clause 7.2.

For 16-QAM the combination given in table 36 shall be used.

Table 36: Code rate combinations for the SDC with 16-QAM (robustness modes A, B, C and D)

Protection level	Rall	R_0	R_1
0	0,5	1/3	2/3

For 4-QAM one of the code rates given in table 37 or table 38 shall be used.

Table 37: Code rate for the SDC with 4-QAM (robustness modes A, B, C and D)

Protection level	Rall	R_0
1	0,5	1/2

Table 38: Code rate for the SDC with 4-QAM (robustness mode E)

Protection level	Rall	R_0
0	0,5	1/2
1	0,25	1/4

Annex J provides the number of input bits per SDC block.

Error detection with a CRC is described in clause 6.

7.5.3 Coding the FAC

The FAC shall use 4-QAM mapping with code rate 0,6 for robustness modes A, B, C and D or 4-QAM mapping with code rate 0,25 for robustness mode E. A fixed code rate shall be applied.

The number of input bits L_{FAC} per FAC block is calculated as given in clause 7.2.

One of the code rates given in table 39 or table 40 shall be used.

Table 39: Code rate for the FAC (robustness modes A, B, C and D)

Rall	R_0
0,6	3/5

Table 40: Code rate for the FAC (robustness mode E)

Rall	R_0
0,25	1/4

Error detection with a CRC is described in clause 6.

7.6 MSC cell interleaving

A cell-wise interleaving shall be applied to the QAM symbols (cells) of the MSC after multilevel encoding. For robustness modes A, B, C and D the possibility to choose low or high interleaving depth (denoted here as short or long interleaving) according to the predicted propagation conditions exists. For robustness mode E only one interleaver depth is available which corresponds to the algorithm for high interleaver depth. The basic interleaver parameters are adapted to the size of a multiplex frame which corresponds to N_{MUX} cells.

For propagation channels below 30 MHz with moderate time-selective behaviour (typical ground wave propagation in LF and MF) the short interleaving provides sufficient time- and frequency diversity for proper operation of the decoding process in the receiver (spreading of error bursts).

The same block interleaving scheme as used for bit interleaving in the multilevel encoder (see clause 7.3.3) is always applied to the N_{MUX} cells of a multiplex frame for all robustness modes.

The input vector of the block interleaver corresponding to the N_{MUX} QAM cells $z_{n,i}$ of multiplex frame *n* is given by:

$$\mathbf{Z}_n = \left(z_{n,0}, z_{n,1}, z_{n,2}, \ldots, z_{n,\mathrm{N}_{\mathrm{MUX}}-1} \right)$$

The output vector with the same number of cells or elements, respectively, is given by:

$$\hat{\mathbf{Z}}_n = \left(\hat{z}_{n,0}, \hat{z}_{n,1}, \hat{z}_{n,2}, \ldots, \hat{z}_{n,\mathrm{N}_{\mathrm{MUX}}-1} \right)$$

where the output elements are selected from the input elements according to:

$$\hat{z}_{n,i} = z_{n,\Pi(i)}$$

The permutation $\Pi(i)$ is obtained from the following relations:

$s = 2^{\lceil \log 2(\mathrm{N_{MUX}}) \rceil}$, $\lceil \ \rceil$ means round towards plus infinity;

$q = s/4 - 1$;

$t_0 = 5$;

$\Pi(0) = 0$;

for $i = 1, 2, \ldots, \mathrm{N_{MUX}} - 1$:

$\Pi(i) = \left(t_0\ \Pi(i-1) + q\right)(\mathrm{mod}\ s)$;

while $\Pi(i) \geq \mathrm{N_{MUX}}$:

$\Pi(i) = \left(t_0\ \Pi(i) + q\right)(\mathrm{mod}\ s)$.

For severe time- and frequency-selective channels below 30 MHz as being typical for signal transmissions in the HF short wave frequency bands and for channels above 30 MHz a greater interleaving depth is provided by an additional convolutional interleaving scheme. For this the interleaving depth D is defined in integer multiples of multiplex frames. As a good trade-off between performance and processing delay a value of D = 5 for robustness modes A, B, C and D and D = 6 for robustness mode E has been chosen.

The output vector for long interleaving with $\mathrm{N_{MUX}}$ cells carrying complex QAM symbols is computed in almost the same way as for short interleaving. The only exception is that the permutation is based not only on the current but also on the last D-1 multiplex frames. The permutation $\Pi(i)$ as defined before is used again to determine the relation between the indices within the output vector $\hat{\mathbf{Z}}_n$ and the D input vectors $\mathbf{Z}_n, \mathbf{Z}_{n-1}, \ldots, \mathbf{Z}_{n-\mathrm{D}+1}$.

The output elements are selected from the input elements according to:

$$\hat{z}_{n,i} = z_{n-\Gamma(i),\Pi(i)}$$

For given value i the selection of the input vector number $n - \Gamma(i)$ for the correspondent element $\Pi(i)$ is done with the following formula:

$$\Gamma(i) = i\ (\mathrm{mod}\,\mathrm{D}) \quad \text{for } i = 0, 1, 2, \ldots, \mathrm{N_{MUX}} - 1$$

Taking into consideration the transmission of the full content of a multiplex frame the overall delay of the pure interleaving/deinterleaving process is given by approximately 2 × 400 ms, i.e. 800 ms, for the short interleaving for robustness modes A, B, C and D. In the case of the long interleaving it corresponds to about 2,4 s for robustness modes A, B, C and D and 0,7 s for robustness mode E.

7.7 Mapping of MSC cells on the transmission super frame structure

The content of M_{TF} consecutive interleaved multiplex frames (with N_{MUX} QAM cells each) is mapped onto a transmission super frame, i.e. the corresponding number N_{SFU} of useful MSC cells is fixed as an integer multiple of M_{TF}. M_{TF} = 3 for robustness modes A, B, C and D and M_{TF} = 4 for robustness mode E. Due to the fact that the number of FAC and synchronization cells is varying from OFDM symbol to OFDM symbol a small loss N_L of 1 or 2 cells can occur compared with the number of available cells in a transmission super frame which is given by:

$$N_{SFA} = N_{SFU} + N_L = M_{TF} \times N_{MUX} + N_L$$

Tables 41 to 45 give the values for the different robustness modes and bandwidths.

Table 41: Number of QAM cells for MSC for robustness mode A

Parameters	Spectrum occupancy					
	0	**1**	**2**	**3**	**4**	**5**
Number of available MSC cells per super frame N_{SFA}	3 778	4 268	7 897	8 877	16 394	18 354
Number of useful MSC cells per super frame N_{SFU}	3 777	4 266	7 896	8 877	16 392	18 354
Number of MSC cells per multiplex frame N_{MUX}	1 259	1 422	2 632	2 959	5 464	6 118
Cell loss per super frame N_L	1	2	1	0	2	0

Table 42: Number of QAM cells for MSC for robustness mode B

Parameters	Spectrum occupancy					
	0	**1**	**2**	**3**	**4**	**5**
Number of available MSC cells per super frame N_{SFA}	2 900	3 330	6 153	7 013	12 747	14 323
Number of useful MSC cells per super frame N_{SFU}	2 898	3 330	6 153	7 011	12 747	14 322
Number of MSC cells per multiplex frame N_{MUX}	966	1 110	2 051	2 337	4 249	4 774
Cell loss per super frame N_L	2	0	0	2	0	1

Table 43: Number of QAM cells for MSC for robustness mode C

Parameters	Spectrum occupancy					
	0	**1**	**2**	**3**	**4**	**5**
Number of available MSC cells per super frame N_{SFA}	-	-	-	5 532	-	11 603
Number of useful MSC cells per super frame N_{SFU}	-	-	-	5 532	-	11 601
Number of MSC cells per multiplex frame N_{MUX}	-	-	-	1 844	-	3 867
Cell loss per super frame N_L	-	-	-	0	-	2

Table 44: Number of QAM cells for MSC for robustness mode D

Parameters	Spectrum occupancy					
	0	**1**	**2**	**3**	**4**	**5**
Number of available MSC cells per super frame N_{SFA}	-	-	-	3 679	-	7 819
Number of useful MSC cells per super frame N_{SFU}	-	-	-	3 678	-	7 818
Number of MSC cells per multiplex frame N_{MUX}	-	-	-	1 226	-	2 606
Cell loss per super frame N_L	-	-	-	1	-	1

Table 45: Number of QAM cells for MSC for robustness mode E

Parameters	Spectrum occupancy					
	0	**1**	**2**	**3**	**4**	**5**
Number of available MSC cells per super frame N_{SFA}	29 842	-	-	-	-	-
Number of useful MSC cells per super frame N_{SFU}	29 840	-	-	-	-	-
Number of MSC cells per multiplex frame N_{MUX}	7 460	-	-	-	-	-
Cell loss per super frame N_L	2	-	-	-	-	-

So the overall data vector for the useful MSC cells in transmission super frame *m* can be described by:

$$\begin{aligned}\mathbf{S}_m &= \left(s_{m,0}, s_{m,1}, s_{m,2}, \ldots, s_{m,N_{SFU-1}}\right) \\ &= \left(\hat{\mathbf{Z}}_{M_{TF}*m}, \hat{\mathbf{Z}}_{M_{TF}*m+1}, \ldots, \hat{\mathbf{Z}}_{M_{TF}*m+M_{TF}-1}\right) \\ &= \left(\hat{z}_{M_{TF}*m,0}, \hat{z}_{M_{TF}*m,1}, \ldots, \hat{z}_{M_{TF}*m,N_{MUX}-1}, \hat{z}_{M_{TF}*m+1,0}, \hat{z}_{M_{TF}*m+1,1}, \ldots, \hat{z}_{M_{TF}*m+1,N_{MUX}-1}, \ldots, \hat{z}_{M_{TF}*m+M_{TF}-1,0}, \hat{z}_{M_{TF}*m+M_{TF}-1,1}, \ldots, \hat{z}_{M_{TF}*m+M_{TF}-1,N_{MUX}-1}\right)\end{aligned}$$

In the case that N_L is unequal to 0 one or two dummy cells, i.e. $\left(\tilde{z}_{m,0}\right)$ or $\left(\tilde{z}_{m,0}, \tilde{z}_{m,1}\right)$, are attached at the end of $\mathbf{S}_m$. Their complex values (i.e. the corresponding QAM symbols) are as defined in table 46.

Table 46: QAM symbols for MSC dummy cells

Number of dummy cells N_L per transmission super frame	Complex values of the dummy cells (QAM symbols)	
	$\tilde{z}_{m,0}$	$\tilde{z}_{m,1}$
1	*a* × (1 + j 1)	
2	*a* × (1 + j 1)	*a* × (1 - j 1)

The value of *a* in table 46 is dependent on the signal constellation chosen for the MSC (see clause 7.4).

8 Transmission structure

8.1 Transmission frame structure and robustness modes

The transmitted signal is organized in transmission super frames.

For robustness modes A, B, C and D, each transmission super frame consists of three transmission frames. For robustness mode E, each transmission super frame consists of four transmission frames.

Each transmission frame has duration T_f, and consists of N_s OFDM symbols.

Each OFDM symbol is constituted by a set of K carriers and transmitted with a duration T_s.

The spacing between adjacent carriers is $1/T_u$.

The symbol duration is the sum of two parts:

- a useful part with duration T_u;
- a guard interval with duration T_g.

The guard interval consists in a cyclic continuation of the useful part, T_u, and is inserted before it.

The OFDM symbols in a transmission frame are numbered from 0 to N_s - 1.

All symbols contain data and reference information.

Since the OFDM signal comprises many separately modulated carriers, each symbol can in turn be considered to be divided into cells, each cell corresponding to the modulation carried on one carrier during one symbol.

An OFDM frame contains:

- pilot cells;
- control cells;
- data cells.

The pilot cells can be used for frame, frequency and time synchronization, channel estimation, and robustness mode identification.

The transmitted signal is described by the following expression:

$$x(t) = \mathrm{Re}\left\{ e^{j2\pi f_R t} \sum_{r=0}^{\infty} \sum_{s=0}^{N_s-1} \sum_{k=K_{\min}}^{K_{\max}} c_{r,s,k}\, \psi_{r,s,k}(t)_{\,k} \right\}$$

where:

$$\psi_{r,s,k}(t) = \begin{cases} e^{j2\pi \frac{k}{T_u}(t - Tg - sT_s - N_s r T_s)} & (s+N_s\ r)T_s \le t \le (s+N_s\ r+1)T_s \\ 0 & otherwise \end{cases}$$

and:

N_s number of OFDM symbols per transmission frame;

k denotes the carrier number (= K_{min}, … , K_{max});

s denotes the OFDM symbol number (= 0 to N_s - 1);

r denotes the transmission frame number (= 0 to infinity);

K is the number of transmitted carriers ($\le K_{max} - K_{min}$);

T_s is the symbol duration;

T_u is the duration of the useful part of a symbol;

T_g is the duration of the guard interval;

f_R is the reference frequency of the RF signal;

$c_{r,s,k}$ complex cell value for carrier k in symbol s of frame number r.

The $c_{r,s,k}$ values depend on the type of cell, as defined below.

For data/control cells (MSC, SDC, FAC), $c_{r,s,k} = z$ where z is the constellation point for each cell as given by the mappings defined in clause 7.

For each reference cell, a defined phase and amplitude is transmitted, $c_{r,s,k} = a_{s,k} U_{s,k}$, where:

$a_{s,k}$ is the amplitude, which always takes one of the values $\{1, \sqrt{2}, 2\}$; and

$U_{s,k} = e^{j2\pi\vartheta_{s,k}}$ is a unit-amplitude term of phase $\vartheta_{s,k}$.

$a_{s,k}$ and $\vartheta_{s,k}$ are defined for each type of reference cell in clause 8.4.

8.2 Propagation-related OFDM parameters

OFDM parameters shall be chosen to match propagation conditions and the coverage area that the operator wants to serve.

Various sets of OFDM parameters are therefore defined for different conditions of propagation and their parameter values are listed in table 47.

Table 47: Numerical values of the OFDM parameters

Robustness mode	Duration T_u	Carrier spacing $1/T_u$	Duration of guard interval T_g	Duration of symbol $T_s = T_u + T_g$	T_g/T_u	Number of symbols per frame N_s
A	24 ms	$41^{2/3}$ Hz	2,66 ms	26,66 ms	1/9	15
B	21,33 ms	$46^{7/8}$ Hz	5,33 ms	26,66 ms	1/4	15
C	14,66 ms	$68^{2/11}$ Hz	5,33 ms	20 ms	4/11	20
D	9,33 ms	$107^{1/7}$ Hz	7,33 ms	16,66 ms	11/14	24
E	2,25 ms	$444^{4/9}$ Hz	0,25 ms	2,5 ms	1/9	40

8.3 Signal bandwidth related parameters

8.3.1 Parameter definition

The OFDM parameters depend upon the available frequency bandwidth, the number of carriers K, and their location with respect to the reference frequency (named DC, in relation with the traditional carrier used in analogue transmissions).

The Spectrum occupancy defines the nominal channel bandwidth. For robustness modes A, B, C and D, the group of carriers carrying the FAC is always to the right (higher in frequency) with respect to the reference frequency, f_R, which is an integer multiple of 1 kHz. For robustness mode E, the group of carriers carrying the FAC is to the left and right (lower respectively higher in frequency) with respect to the reference frequency, f_R, which is an integer multiple of 10 kHz. Table 48 relates the spectrum occupancy parameter, signalled in the FAC (see clause 6.3), to the nominal channel bandwidth, and figures 31 and 32 show the position of the carriers for $f_R < 30$ MHz.

Table 48: Relationship between spectrum occupancy parameter and channel bandwidth

	Spectrum occupancy					
	0	1	2	3	4	5
Channel bandwidth (kHz) robustness modes A, B, C and D	4,5	5	9	10	18	20
Channel bandwidth (kHz) robustness mode E	100	-	-	-	-	-

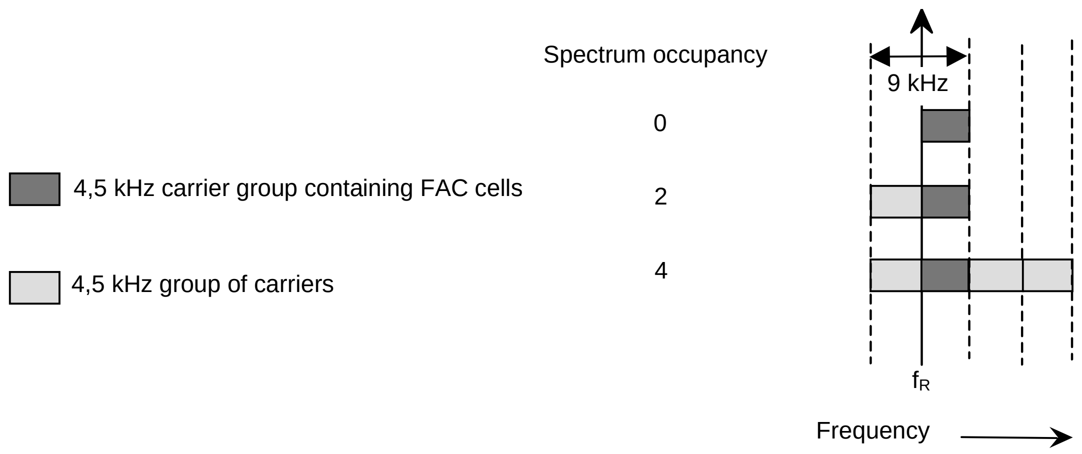


Figure 31: Spectrum occupancy for 9 kHz channels

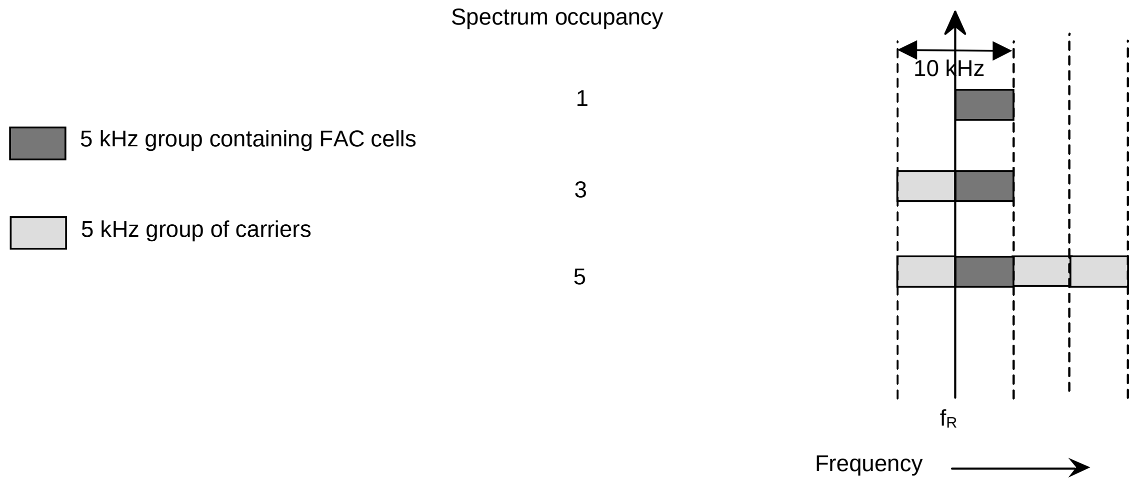


Figure 32: Spectrum occupancy for 10 kHz channels

The carriers are indexed by $k \in [K_{min}, K_{max}]$, $k = 0$ being the DC carrier and determined by the following values depending on the choice made and related to the occupied bandwidth.

Carriers with $k < 0$ are said to be to the left of DC, and $k > 0$, to the right of DC.

Table 49 presents the lower and upper carrier numbers for each robustness mode and nominal bandwidth.

Table 49: Carrier numbers for each robustness mode

Robustness mode	Carrier	Spectrum occupancy					
		0	1	2	3	4	5
A	Kmin	2	2	-102	-114	-98	-110
	Kmax	102	114	102	114	314	350
B	Kmin	1	1	-91	-103	-87	-99
	Kmax	91	103	91	103	279	311
C	Kmin	-	-	-	-69	-	-67
	Kmax	-	-	-	69	-	213
D	Kmin	-	-	-	-44	-	-43
	Kmax	-	-	-	44	-	135
E	Kmin	-106	-	-	-	-	-
	Kmax	106	-	-	-	-	-

The DC carrier and certain carriers around DC are not used in some robustness modes, as detailed in table 50.

Table 50: Unused carriers according to robustness mode

Robustness mode	Unused carrier(s)
A	$k \in \{-1,0,1\}$
B	$k \in \{0\}$
C	$k \in \{0\}$
D	$k \in \{0\}$
E	none

8.3.2 Simulcast transmission

For robustness modes A, B, C and D, the DRM signal is designed to work in the same broadcast bands as AM signals. Simulcast transmission of services using DRM and AM can be performed by the juxtaposition of the analogue AM signal (DSB or VSB or SSB) and a DRM digital signal. Many arrangements are possible and some are illustrated in annex K.

The spectrum occupancy number relates to the DRM signal. A broadcaster may choose to signal the presence of the AM simulcast by use of the Alternative frequency signalling: Other services data entity in the SDC (see clause 6.4.3.12).

8.4 Pilot cells

8.4.1 Functions and derivation

Some cells within the OFDM transmission frame are modulated with known fixed phases and amplitudes.

These cells are pilot cells for channel estimation and synchronization. The positions, amplitudes and phases of these cells are carefully chosen to optimize the performance, especially the initial synchronization duration and reliability.

The phases are defined, directly or indirectly, in 1 024$^{\text{ths}}$ of a cycle, i.e. $U_{s,k} = e^{j2\pi\vartheta_{s,k}} = e^{\frac{j2\pi\vartheta_{1024}[s,k]}{1024}}$, where $\vartheta_{1024}[s,k]$ takes integer values and is either explicitly tabulated or derived using integer arithmetic, as defined in the following clauses (clause 8.4.2 to clause 8.4.5.2 inclusive).

8.4.2 Frequency references

8.4.2.0 Introduction

These cells are used by the receiver to detect the presence of the received signal and to estimate its frequency offset. They may also be used for channel estimation and various tracking processes.

For robustness mode E, no frequency reference cells are defined.

8.4.2.1 Cell positions

For robustness modes A, B, C and D, frequency references are located at frequencies which are common to these four robustness modes.

There are three frequency references, which are 750 Hz, 2 250 Hz and 3 000 Hz as referenced to the DC carrier, as defined in table 51.

Table 51: Carrier numbers for frequency references

Robustness mode	Carrier numbers
A	18, 54, 72
B	16, 48, 64
C	11, 33, 44
D	7, 21, 28
E	none

They shall be present in all symbols of each transmission frame.

8.4.2.2 Cell gains and phases

For robustness modes A, B, C and D, all frequency reference cells shall have a power gain of 2, i.e. $a_{s,k} = \sqrt{2}$, in order to optimize performances at low signal to noise ratio and be compatible when the same cell functions as both a frequency reference and a time reference.

The phases are defined as follows. For the first symbol in the frame (i.e. $s = 0$), the phases $\vartheta_{1024}[s,k]$ are given in table 52.

Table 52: Cell phases for frequency references

Robustness mode	Carrier index, k	Phase index, $\vartheta_{1024}[0,k]$
A	18	205
	54	836
	72	215
B	16	331
	48	651
	64	555
C	11	214
	33	392
	44	242
D	7	788
	21	1 014
	28	332
E	none	none

For subsequent symbols, the phases are chosen in order to ensure the tones are continuous, which is achieved by applying the following rules.

For robustness modes A, B and C, and carrier 28 only of robustness mode D:

$$\vartheta_{1024}[s,k] = \vartheta_{1024}[0,k]$$

For robustness mode D, carriers 7 and 21:

$$\vartheta_{1024}[s,k] = \vartheta_{1024}[0,k] \text{, for even values of } s \text{, and}$$

$$\vartheta_{1024}[s,k] = \left(\vartheta_{1024}[0,k] + 512\right) \bmod 1024 \text{, for odd values of } s.$$

NOTE: This is equivalent to the complex value $U_{s,k}$ multiplied by - 1 for odd values of s.

8.4.3 Time references

8.4.3.0 Introduction

These cells are located in the first OFDM symbol of each transmission frame, i.e. $s = 0$.

The time reference cells are mainly used for performing ambiguity resolution since guard time correlation provides a fast and frequency insensitive estimation of time of arrival with a periodicity of one symbol. They are used for determining the first symbol of a transmission frame. They can also be used for frequency-offset estimation.

8.4.3.1 Cell positions and phases

Tables 53 to 57 define the phases of the time reference cells, and the phases of the frequency reference cells for the first symbol of the transmission frame.

$\vartheta_{1024}[0,k]$ is the phase index in 1 024$^{\text{ths}}$ of a cycle.

Table 53: Phase of time reference cells for robustness mode A

Carrier index, k	Phase index, $\vartheta_{1024}[0,k]$
17	973
18*	205
19	717
21	264
28	357
29	357
32	952
33	440
39	856
40	88
41	88
53	68
54*	836
55	836
56	836
60	1 008
61	1 008
63	752
71	215
72*	215
73	727
NOTE: Carrier numbers marked with an asterisk "*" also serve as frequency references (see clause 8.4.2.1); the definitions of phase index are consistent.	

Table 54: Phase of time reference cells for robustness mode B

Carrier index k	Phase index, $\vartheta_{1024}[0,k]$
14	304
16*	331
18	108
20	620
24	192
26	704
32	44
36	432
42	588
44	844
48*	651
49	651
50	651
54	460
56	460
62	944
64*	555
66	940
68	428
NOTE: Carrier numbers marked with an asterisk "*" also serve as frequency references (see clause 8.4.2.1); the definitions of phase index are consistent.	

Table 55: Phase of time reference cells for robustness mode C

Carrier index k	Phase index, $\vartheta_{1024}[0,k]$
8	722
10	466
11*	214
12	214
14	479
16	516
18	260
22	577
24	662
28	3
30	771
32	392
33*	392
36	37
38	37
42	474
44*	242
45	242
46	754
NOTE: Carrier numbers marked with an asterisk "*" also serve as frequency references (see clause 8.4.2.1); the definitions of phase index are consistent.	

Table 56: Phase of time reference cells for robustness mode D

Carrier index k	Phase index, $\vartheta_{1024}[0,k]$
5	636
6	124
7*	788
8	788
9	200
11	688
12	152
14	920
15	920
17	644
18	388
20	652
21*	1 014
23	176
24	176
26	752
27	496
28*	332
29	432
30	964
32	452
NOTE: Carrier numbers marked with an asterisk "*" also serve as frequency references (see clause 8.4.2.1); the definitions of phase index are consistent.	

Table 57: Phase of time reference cells for robustness mode E

Carrier index, k	Phase index, $\vartheta_{1024}[0,k]$
-80	219
-79	475
-77	987
-53	652
-52	652
-51	140
-32	819
-31	819
12	907
13	907
14	651
21	903
22	391
23	903
40	203
41	203
42	203
67	797
68	29
79	508
80	508

8.4.3.2 Cell gains

All time reference cells have a power gain of 2,0 in order to optimize performance at low signal to noise ratio, i.e. $a_{s,k} = \sqrt{2}$.

8.4.4 Gain references

8.4.4.0 Introduction

The gain reference cells are mainly used for coherent demodulation. These cells are scattered throughout the overall time frequency pattern and are used by the receiver to estimate the channel response.

8.4.4.1 Cell positions

In a transmission frame, for the symbol of index s (ranging from 0 to N_s - 1), carriers for which index k belongs to the subsets as defined in table 58 are gain references.

Table 58: Carrier indices k for gain reference cells

Robustness mode	Subset	Condition	Periodicity of the gain reference pattern
A	k = 2 + 4 × (s mod 5) + 20 × p	p integer $k_{\min} \le k \le k_{\max}$	5 symbols
B	k = 1 + 2 × (s mod 3) + 6 × p	p integer $k_{\min} \le k \le k_{\max}$	3 symbols
C	k = 1 + 2 × (s mod 2) + 4 × p	p integer $k_{\min} \le k \le k_{\max}$	2 symbols
D	k = 1 + (s mod 3) + 3 × p	p integer $k_{\min} \le k \le k_{\max}$	3 symbols
E	k = 2 + 4 × (s mod 4) + 16 × p	p integer $k_{\min} \le k \le k_{\max}$	4 symbols

NOTE: The gain reference cell patterns have been chosen such that the edge carriers are included as gain reference cell positions.

Annex L gives some example figures illustrating the position of the gain reference cells.

8.4.4.2 Cell gains

Gain reference cells mostly have a power gain of 2 (i.e. $a_{s,k} = \sqrt{2}$), in order to optimize performances at low signal to noise ratio. However, gain reference cells close to the band lower and upper edges are over-boosted by a further power gain of 2 (i.e. overall power gain of 4, so that the amplitude $a_{s,k} = 2$) as defined in table 59.

Table 59: Carrier numbers given a power boost of 4, i.e. $a_{s,k} = 2$

Robustness mode	Spectrum occupancy					
	0	1	2	3	4	5
A	2, 6, 98, 102	2, 6, 110, 114	-102, -98, 98, 102	-114, -110, 110, 114	-98, -94, 310, 314	-110, -106, 346, 350
B	1, 3, 89, 91	1, 3, 101, 103	-91, -89, 89, 91	-103, -101, 101, 103	-87, -85, 277, 279	-99, -97, 309, 311
C	-	-	-	-69, -67, 67, 69	-	-67, -65, 211, 213
D	-	-	-	-44, -43, 43, 44	-	-43, -42, 134, 135
E	-106, -102, 102, 106	-	-	-	-	-

8.4.4.3 Cell phases

8.4.4.3.0 Introduction

In some cases gain references fall in locations which coincide with those already defined for either frequency or time references. In these cases, the phase definitions given in clauses 8.4.2 and 8.4.3 take precedence.

In all other locations, the phases of the gain reference cells are obtained by integer arithmetic applied to small tables of values, as defined in the following procedure.

8.4.4.3.1 Procedure for calculation of cell phases

The procedure is:

First, compute values of *m*, *n* and *p* for each cell, where the carrier number is *k* and the symbol number is *s*:

$$n = s \bmod y,$$
$$m = \lfloor s / y \rfloor$$
$$p = \frac{k - k_0 - nx}{xy}$$

x, *y*, and k_0 are constants which are defined for each robustness mode in table 60.

Table 60: Definition of x, y, k_0

Robustness mode	x	y	k_0
A	4	5	2
B	2	3	1
C	2	2	1
D	1	3	1
E	4	4	2

NOTE 1: The value of *p* obtained by this procedure is an integer, as a consequence of the definition of reference cell locations in clause 8.4.4.1; while the values of *n* and *m* are integer by definition of the mathematical operations producing them.

Secondly, calculate for robustness modes A, B, C and D the integer phase index by the following formula:

$$\vartheta_{1024}[s,k] = \left(4Z_{256}[n,m] + pW_{1024}[n,m] + p^2(1+s)Q_{1024}\right) \bmod 1024$$; or

calculate for robustness mode E the integer phase index by the following formula:

$$\vartheta_{1024}[s,k] = (p^2 R_{1024}[n,m] + pZ_{1024}[n,m] + Q_{1024}[n,m]) \bmod 1024$$

Q_{1024} and the small tables $Z_{256}[n,m]$, $W_{1024}[n,m]$, $R_{1024}[n,m]$, $Z_{1024}[n,m]$ and $Q_{1024}[n,m]$ are defined for each robustness mode in clauses 8.4.4.3.2 to 8.4.4.3.6.

NOTE 2: The values in table $Z_{256}[n,m]$ may be represented precisely as 8-bit unsigned integers; Q_{1024}, $W_{1024}[n,m]$, $R_{1024}[n,m]$, $Z_{1024}[n,m]$ and $Q_{1024}[n,m]$ may be represented precisely as 10-bit unsigned integers.

8.4.4.3.2 Robustness mode A

The $W_{1024}[n,m]$ matrix is defined as:

$W_{1024}[n,m]$ = {	{228,	341,	455},
	{455,	569,	683},
	{683,	796,	910},
	{910,	0,	114},
	{114,	228,	341}}

The $Z_{256}[n,m]$ matrix is defined as:

$Z_{256}[n,m]$ = {	{0,	81,	248},
	{18,	106,	106},
	{122,	116,	31},
	{129,	129,	39},
	{33,	32,	111}}

Q_{1024} = 36.

8.4.4.3.3 Robustness mode B

The $W_{1024}[n,m]$ matrix is defined as:

$W_{1024}[n,m]$ = {	{512,	0,	512,	0,	512},
	{0,	512,	0,	512,	0},
	{512,	0,	512,	0,	512}}

The $Z_{256}[n,m]$ matrix is defined as:

$Z_{256}[n,m]$ = {	{0,	57,	164,	64,	12},
	{168,	255,	161,	106,	118},
	{25,	232,	132,	233,	38}}

Q_{1024} = 12.

8.4.4.3.4 Robustness mode C

The $W_{1024}[n,m]$ matrix is defined as:

$W_{1024}[n,m]$ = {	{465,	372,	279,	186,	93,	0,	931,	838,	745,	652},
	{931,	838,	745,	652,	559,	465,	372,	279,	186,	93}}

The $Z_{256}[n,m]$ matrix is defined as:

$Z_{256}[n,m]$ = {	{0,	76,	29,	76,	9,	190,	161,	248,	33,	108},
	{179,	178,	83,	253,	127,	105,	101,	198,	250,	145}}

Q_{1024} = 12.

8.4.4.3.5 Robustness mode D

The $W_{1024}[n,m]$ matrix is defined as:

$W_{1024}[n,m]$ = {	{366,	439,	512,	585,	658,	731,	805,	878},
	{731,	805,	878,	951,	0,	73,	146,	219},
	{73,	146,	219,	293,	366,	439,	512,	585}}

The $Z_{256}[n,m]$ matrix is defined as:

$Z_{256}[n,m]$ = {	{0,	240,	17,	60,	220,	38,	151,	101},
	{110,	7,	78,	82,	175,	150,	106,	25},
	{165,	7,	252,	124,	253,	177,	197,	142}}

Q_{1024} = 14.

8.4.4.3.6 Robustness mode E

The $R_{1024}\ [n,m]$ matrix is defined as:

$R_{1024}\ [n,m]$ = {	{39,	118,	197,	276,	354,	433,	39,	118,	197,	276},
	{37,	183,	402,	37,	183,	402,	37,	183,	402,	37},
	{110,	329,	475,	110,	329,	475,	110,	329,	475,	110},
	{79,	158,	236,	315,	394,	473,	79,	158,	236,	315}}

The $Z_{1024}\ [n,m]$ matrix is defined as:

$Z_{1024}\ [n,m]$ = {	{473,	394,	315,	236,	158,	79,	0,	0,	0,	0},
	{183,	914,	402,	37,	475,	841,	768,	768,	987,	183},
	{549,	622,	475,	110,	37,	622,	256,	768,	329,	549},
	{79,	158,	236,	315,	394,	473,	158,	315,	473,	630}}

The $Q_{1024}\ [n,m]$ matrix is defined as:

$Q_{1024}\ [n,m]$ = {	{329,	489,	894,	419,	607,	519,	1 020,	942,	817,	939},
	{824,	1 023,	74,	319,	225,	207,	348,	422,	395,	92},
	{959,	379,	7,	738,	500,	920,	440,	727,	263,	733},
	{907,	946,	924,	91,	189,	133,	910,	804,	1 022,	433}}

8.4.5 AFS references

8.4.5.0 Introduction

The AFS reference cells are only provided in robustness mode E. These cells are located in the fifth OFDM symbol, i.e. s = 4, of the first transmission frame and the fortieth symbol, i.e. s = 39, of the fourth transmission frame.

The AFS reference cells are mainly used to improve the channel estimation in the AFS case and to make "snooping" at another frequency more reliable. The AFS reference cells of OFDM symbol s = 39 in the fourth transmission frame make it possible to complete the channel estimation in a proper way before switching to the alternative frequency. The AFS reference cells of OFDM symbol s = 4 in the first transmission frame are part of the SDC symbols and help to improve the channel estimation on return to the original frequency.

8.4.5.1 Cell positions and phases

Table 61 defines the positions and phases of the AFS reference cells, for the fifth OFDM symbol i.e. s = 4 of the first transmission frame and the fortieth symbol i.e. s = 39 of the fourth transmission frame.

Table 61: Phase of AFS reference cells for robustness mode E

Carrier index k	Phase index, $\vartheta_{1024}[0,k]$ s = 4	Phase index, $\vartheta_{1024}[0,k]$ s = 39
-106	134	115
-102	866	135
-98	588	194*
-94	325*	293
-90	77	431
-86	868	608
-82	649	825*
-78	445*	57
-74	256	353
-70	82	688
-66	946	38*
-62	801*	452
-58	671	905
-54	556	373
-50	455	905*
-46	369*	452
-42	298	39
-38	242	689
-34	200	354*
-30	173*	59
-26	161	827
-22	164	610
-18	181	433*
-14	213*	295
-10	260	197
-6	322	138
-2	398	118*
2	489*	138
6	595	197
10	716	295
14	851	433*
18	1 001*	610
22	142	827
26	322	59
30	516	354*
34	725*	689
38	949	39
42	164	452
46	417	905*
50	685*	373
54	968	905
58	242	452
62	554	38*
66	881*	688
70	199	353
74	556	57
78	927	825*
82	289*	608
86	690	431

Carrier index k	Phase index, $\vartheta_{1024}[0,k]$ s = 4	Phase index, $\vartheta_{1024}[0,k]$ s = 39
90	82	293
94	512	194*
98	957*	135
102	393	115
106	868	134
NOTE: Carrier numbers belonging to phase indexes marked with an asterisk "*" serve as AFS and gain references (see clause 8.4.4.1); the definitions of phase index are consistent; the amplitude definition is that one of the gain reference.		

8.4.5.2 Cell gains

All AFS reference cells have a power gain of 1,0 i.e. the AFS reference cells are not boosted.

8.5 Control cells

8.5.1 General

The control cells consist of two parts:

- the Fast Access Channel (FAC), integrated in every transmission frame. It is used to quickly obtain the necessary information for the receiver to be able to demodulate the DRM signal;
- the Service Description Channel (SDC), repeated every transmission super frame. It contains all the additional information that describes the services currently available. The SDC is also used for Alternative Frequency Switching (AFS).

Figure 33 describes the time-frequency location of these signals.

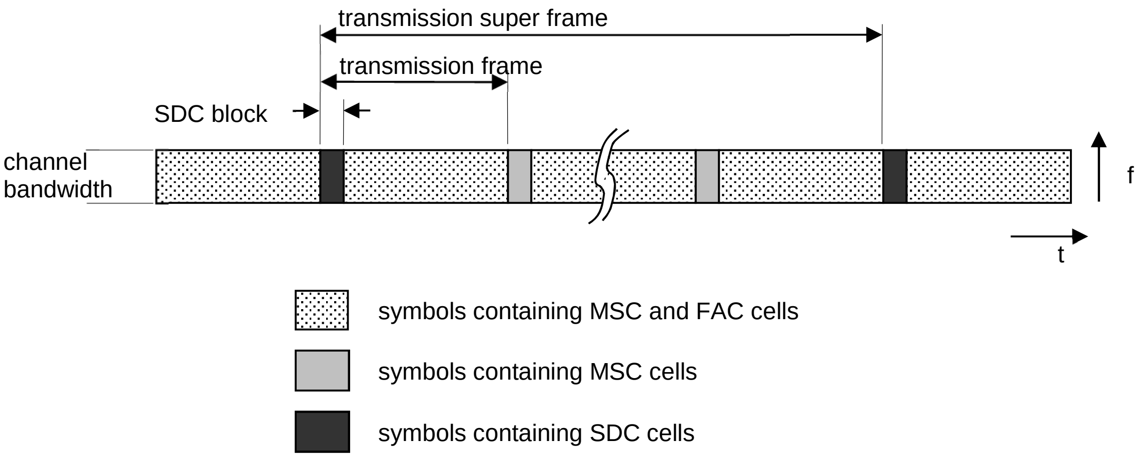


Figure 33: Time-frequency location of FAC and SDC signals

8.5.2 FAC cells

8.5.2.1 Cell positions

The cells used for FAC are cells that are neither frequency references, nor time references, nor gain references, nor data cells in the symbols that do not contain the SDC.

FAC cells convey highly protected QAM symbols that allow fast detection by the receiver of the type of signal it is currently receiving.

For robustness modes A, B, C and D there are 65 FAC cells and for robustness mode E there are 244 FAC cells. Tables 62 to 66 give the position of the FAC cells for each robustness mode.

Table 62: Position of the FAC cells in robustness mode A

Symbol	Carrier number
0	
1	
2	26, 46, 66, 86
3	10, 30, 50, 70, 90
4	14, 22, 34, 62, 74, 94
5	26, 38, 58, 66, 78
6	22, 30, 42, 62, 70, 82
7	26, 34, 46, 66, 74, 86
8	10, 30, 38, 50, 58, 70, 78, 90
9	14, 22, 34, 42, 62, 74, 82, 94
10	26, 38, 46, 66, 86
11	10, 30, 50, 70, 90
12	14, 34, 74, 94
13	38, 58, 78
14	

Table 63: Position of the FAC cells in robustness mode B

Symbol	Carrier number
0	
1	
2	13, 25, 43, 55, 67
3	15, 27, 45, 57, 69
4	17, 29, 47, 59, 71
5	19, 31, 49, 61, 73
6	9, 21, 33, 51, 63, 75
7	11, 23, 35, 53, 65, 77
8	13, 25, 37, 55, 67, 79
9	15, 27, 39, 57, 69, 81
10	17, 29, 41, 59, 71, 83
11	19, 31, 43, 61, 73
12	21, 33, 45, 63, 75
13	23, 35, 47, 65, 77
14	

Table 64: Position of the FAC cells in robustness mode C

Symbol	Carrier number
0	
1	
2	
3	9, 21, 45, 57
4	23, 35, 47
5	13, 25, 37, 49
6	15, 27, 39, 51
7	5, 17, 29, 41, 53
8	7, 19, 31, 43, 55
9	9, 21, 45, 57
10	23, 35, 47
11	13, 25, 37, 49
12	15, 27, 39, 51
13	5, 17, 29, 41, 53
14	7, 19, 31, 43, 55
15	9, 21, 45, 57
16	23, 35, 47
17	13, 25, 37, 49
18	15, 27, 39, 51
19	

Table 65: Position of the FAC cells in robustness mode D

Symbol	Carrier number
0	
1	
2	
3	9, 18, 27
4	10, 19
5	11, 20, 29
6	12, 30
7	13, 22, 31
8	5, 14, 23, 32
9	6, 15, 24, 33
10	16, 25, 34
11	8, 17, 26, 35
12	9, 18, 27, 36
13	10, 19, 37
14	11, 20, 29
15	12, 30
16	13, 22, 31
17	5, 14, 23, 32
18	6, 15, 24, 33
19	16, 25, 34
20	8, 17, 26, 35
21	9, 18, 27, 36
22	10, 19, 37
23	

Table 66: Position of the FAC cells in robustness mode E

Symbol	Carrier number
0	
1	
2	
3	
4	
5	-78, -62, -46, -30, -14, 2, 18, 34, 50, 66, 82
6	-90, -74, -58, -42, -26, -10, 6, 22, 38, 54, 70, 86
7	-86, -70, -54, -38, -22, -6, 10, 26, 42, 58, 74, 90
8	-82, -66, -50, -34, -18, -2, 14, 30, 46, 62, 78
9	-78, -62, -46, -30, -14, 2, 18, 34, 50, 66, 82
10	-90, -74, -58, -42, -26, -10, 6, 22, 38, 54, 70, 86
11	-86, -70, -54, -38, -22, -6, 10, 26, 42, 58, 74, 90
12	-82, -66, -50, -34, -18, -2, 14, 30, 46, 62, 78
13	-78, -62, -46, -30, -14, 2, 18, 34, 50, 66, 82
14	-90, -74, -58, -42, -26, -10, 6, 22, 38, 54, 70, 86
15	-86, -70, -54, -38, -22, -6, 10, 26, 42, 58, 74, 90
16	-82, -66, -50, -34, -18, -2, 14, 30, 46, 62, 78
17	-78, -62, -46, -30, -14, 2, 18, 34, 50, 66, 82
18	-90, -74, -58, -42, -26, -10, 6, 22, 38, 54, 70, 86
19	-86, -70, -54, -38, -22, -6, 10, 26, 42, 58, 74, 90
20	-82, -66, -50, -34, -18, -2, 14, 30, 46, 62, 78
21	-78, -62, -46, -30, -14, 2, 18, 34, 50, 66, 82
22	-90, -74, -58, -42, -26, -10, 6, 22, 38, 54, 70, 86
23	-86, -70, -54, -38, -22, -6, 10, 26, 42, 58, 74, 90
24	-82, -66, -50, -34, -18, -2, 14, 30, 46, 62, 78
25	-78, -62, -46, -30, -14, 2, 18, 34, 50, 66, 82
26	-90, -74, -58
27	
28	
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	

8.5.2.2 Cell gains and phases

The $c_{r,s,k}$ values are normalized modulation values of the constellation point z according to the modulation alphabet used for the FAC (4-QAM) (see figure 30).

Successive constellation points are assigned to the FAC cells of a transmission frame in order of increasing carrier index *k*, starting from the most negative *k*; then in time order starting from the first FAC bearing symbol of the frame.

8.5.3 SDC cells

8.5.3.1 Cell positions

The cells used for SDC are all the cells in the SDC symbols which are neither frequency references, nor time references, nor gain references for which $k_{\min} \leq k \leq k_{\max}$ and *k* does not belong to the subset of unused carriers defined above.

For robustness modes A and B, the SDC symbols are symbols 0 and 1 of each transmission super frame. For robustness modes C and D, the SDC symbols are symbols 0, 1 and 2 of each transmission super frame. For robustness mode E, the SDC symbols are symbols 0, 1, 2, 3 and 4 of each transmission super frame.

8.5.3.2 Cell gains and phases

The $c_{r,s,k}$ values are normalized modulation values of the constellation point z according to the modulation alphabet used for the SDC (16- or 4-QAM for robustness modes A, B, C and D and 4-QAM for robustness mode E, see figures 29 and 30).

Successive constellation points are assigned to the SDC cells of a transmission super frame in order of increasing carrier index k, starting from the most negative k; then in time order starting from the first SDC bearing symbol of the super frame.

8.6 Data cells

8.6.1 Cell positions

Data cells are all cells which are neither pilot cells, nor control cells; for which $k_{\min} \le k \le k_{\max}$ and k does not belong to the subset of unused carriers defined above.

8.6.2 Cell gains and phases

The $c_{r,s,k}$ values are the normalized modulation values of the constellation point z according to the modulation alphabet used for the MSC (64-QAM or 16-QAM for robustness modes A, B, C and D and 16-QAM or 4-QAM for robustness mode E, see figures 26 to 30) taken from the vector S_m (see clause 7.7).

Successive elements $s_{m,i}$ are assigned to the cells of a transmission super frame in order of increasing carrier index k, starting from the most negative k; then in time order starting from the first non-SDC symbol of the super frame.

Annex A (informative): Simulated system performance

This annex provides simulated system performance anticipating perfect channel estimation, ideal synchronization and the absence of phase noise and quantization effects. The signal power includes pilots and the guard interval. Channel decoding is assumed to be done with single stage Viterbi decoding for 4-QAM modulation and with a multistage decoder with two iterations for 64-QAM modulation.

The results in table A.1 are given for five of the channels of clause B.1, whereby the associated robustness modes are A for channels 1 and 2, and B for channels 3 to 5. The associated code rate is R = 0,6 and the modulation is 64-QAM.

Table A.1: Required S/N for a transmission to achieve a BER = 1 × 10^{-4} after the channel decoder for the MSC(Mode A/B)

Channel model	C/N
Channel 1	14,9 dB
Channel 2	16,5 dB
Channel 3	23,2 dB
Channel 4	22,3 dB
Channel 5	20,4 dB

Further results for other combinations of DRM transmission and service parameters (including real channel estimation behaviour of the receiver) can be found in Recommendation ITU-R BS.1615 [i.1].

The results in table A.2 are given for six of the channels of clause B.2, whereby the associated robustness mode is E. The code rate is R=0,33 and the modulation is 4-QAM.

Table A.2: Required C/N for a transmission to achieve a BER = 1 × 10^{-4} after the channel decoder for the MSC (Mode E)

Channel model	C/N
Channel 7 (AWGN)	1,3 dB
Channel 8 (Urban) at 60 km/h	7,3 dB
Channel 9 (Rural)	5,6 dB
Channel 10 (Terrain obstructed)	5,4 dB
Channel 11 (Hilly terrain)	5,5 dB
Channel 12 (SFN)	5,4 dB

The results in table A.3 are given for six of the channels of clause B.2, whereby the associated robustness mode is E. The code rate is R=0,5 and the modulation is 16-QAM.

Table A.3: Required C/N for a transmission to achieve a BER = 1 × 10^{-4} after the channel decoder for the MSC (Mode E)

Channel model	C/N
Channel 7 (AWGN)	7,9 dB
Channel 8 (Urban) at 60 km/h	15,4 dB
Channel 9 (Rural)	13,1 dB
Channel 10 (Terrain obstructed)	12,6 dB
Channel 11 (Hilly terrain)	12,8 dB
Channel 12 (SFN)	12,3 dB

Annex B (informative): Definition of channel profiles

B.1 Robustness modes A, B, C and D

The channels to be considered are the LF, MF and HF broadcast radio transmission channels. In principle all three are multipath channels because the surface of the earth and the ionosphere are involved in the mechanism of electromagnetic wave propagation.

The approach is to use stochastic time-varying models with a stationary statistics and define models for good, moderate and bad conditions by taking appropriate parameter values of the general model. One of those models with adaptable parameters is the Wide Sense Stationary Uncorrelated Scattering model (WSSUS model). The justification for the stationary approach with different parameter sets is, that results on real channels lead to BER curves between best and worst cases found in the simulation.

The channel models have been generated from the following equations where e(t) and s(t) are the complex envelopes of the input and output signals respectively:

$$s(t) = \sum_{k=1}^{n} \rho_k c_k(t) e(t - \Delta_k) \qquad \text{(B.1)}$$

This is a tapped delay-line where:

- ρ_k is the attenuation of the path number k - listed in table B.1.
- Δ_k is the relative delay of the path number k - listed in table B.1.
- the time-variant tap weights $\{c_k(t)\}$ are zero mean complex-valued stationary Gaussian random processes. The magnitudes $|c_k(t)|$ are Rayleigh-distributed and the phases Φ (t) are uniformly distributed.

For each weight $\{c_k(t)\}$ there is one stochastic process, characterized by its variance and its Power Density Spectrum (PDS). The variance is a measure for the average signal power which is received via this path and is defined by the relative attenuation ρ_k - listed in table B.1 - and the PDS determines the average speed of variation in time. The width of the PDS is quantified by a number and is referred to as the Doppler spread D_{sp} of that path - listed in table B.1.

There might be also a non-zero centre frequency of the PDS, which can be interpreted as an average frequency shift or Doppler shift D_{sh} - listed in table B.1.

The PDS is modelled by filtering of white noise (i.e. with constant PDS) and is equal to:

$$\phi_{n_t n_t}(f) = N_0 \cdot |H(f)|^2 \qquad \text{(B.2)}$$

H(f) is the transfer function of the filter. The stochastic processes belonging to every individual path then become Rayleigh processes. For the ionospheric path, a Gaussian shape has proven to be a good approach with respect to real observations.

The Doppler profile on each path k is then defined as:

$$|H(f)|^2 = \frac{1}{\sqrt{2\pi\sigma_d^2}} e^{-\frac{(f - Dsh)^2}{2\sigma_d^2}} \qquad \text{(B.3)}$$

The Doppler spread is specified as 2-sided and contains 68 % of the power:

$$D_{sp} = 2\,\sigma_d \qquad \text{(B.4)}$$

Table B.1: Set of channels

Channel no 1: AWGN		**good typical/moderate bad**		**LF, MF, HF LF, var.SNR**
	path 1	**path 2**	**path 3**	**path 4**
Delay (Δ_k)	0			
Path gain, rms (ρ_k)	1			
Doppler shift (D_{sh})	0			
Doppler spread (D_{sp})	0			

Channel no 2: Rice with delay		**good typical/moderate bad**		**MF, HF**
	path 1	**path 2**	**path 3**	**path 4**
Delay (Δ_k)	0	1 ms		
Path gain, rms (ρ_k)	1	0,5		
Doppler shift (D_{sh})	0	0		
Doppler spread (D_{sp})	0	0,1 Hz		

Channel no 3: US Consortium		**good typical/moderate bad**		**HF MF**
	path 1	**path 2**	**path 3**	**path 4**
Delay (Δ_k)	0	0,7 ms	1,5 ms	2,2 ms
Path gain, rms (ρ_k)	1	0,7	0,5	0,25
Doppler shift (D_{sh})	0,1 Hz	0,2 Hz	0,5 Hz	1,0 Hz
Doppler spread (D_{sp})	0,1 Hz	0,5 Hz	1,0 Hz	2,0 Hz

Channel no 4: CCIR Poor		**good typical/moderate bad**		**HF**
	path 1	**path 2**	**path 3**	**path 4**
Delay (Δ_k)	0	2 ms		
Path gain, rms (ρ_k)	1	1		
Doppler shift (D_{sh})	0	0		
Doppler spread (D_{sp})	1 Hz	1 Hz		

Channel no 5		**good typical/moderate bad**		**HF**
	path 1	**path 2**	**path 3**	**path 4**
Delay (Δ_k)	0	4 ms		
Path gain, rms (ρ_k)	1	1		
Doppler shift (D_{sh})	0	0		
Doppler spread (D_{sp})	2 Hz	2 Hz		

Channel no 6		good typical/moderate bad		HF
	path 1	path 2	path 3	path 4
Delay (Δ_k)	0	2 ms	4 ms	6 ms
Path gain, rms (ρ_k)	0,5	1	0,25	0,0625
Doppler shift (D_{sh})	0	1,2 Hz	2,4 Hz	3,6 Hz
Doppler spread (D_{sp})	0,1 Hz	2,4 Hz	4,8 Hz	7,2 Hz

B.2 Robustness mode E

In contrast to the lower frequency bands the radio wave propagation in the VHF bands is characterized by diffraction, scattering and reflection of the electromagnetic waves on their way between the transmitter and the receiver. Typically the waves arrive at different times at the receiver (multipath propagation) resulting in more or less strong frequency-selective fading (dependent on system bandwidth). In addition movements of the receiver or surrounding objects cause a time variation of the channel characteristic (Doppler effect). In contrast to sky wave propagation e.g. at short waves ionospheric variations play no role for channel modelling for the VHF bands.

The approach is to use stochastic time-varying models with a stationary statistics and define models for good, moderate and bad conditions by taking appropriate parameter values of the general model. One of those models with adaptable parameters is the Wide Sense Stationary Uncorrelated Scattering model (WSSUS model). The justification for the stationary approach with different parameter sets is that results on real channels lead to BER curves between best and worst cases found in the simulation.

Additional variations of the short-term average power (slow or lognormal fading) caused by changing environment (e.g. building structure) or phenomena like sporadic E layer propagation are not incorporated in the WSSUS model. Their effects, as well as the influence of disturbances like man-made noise, are normally integrated in the computation of the coverage probability during the network planning process.

The channel models have been generated from the following equations where e(t) and s(t) are the complex envelopes of the input and output signals respectively:

$$s(t) = \sum_{k=1}^{n} \rho_k c_k(t) e(t - \Delta_k) \tag{B.5}$$

This is a tapped delay-line where:

- ρ_k is the attenuation of the path number k - listed in table B.2.
- Δ_k is the relative delay of the path number k - listed in table B.2.
- the time-variant tap weights $\{c_k(t)\}$ are zero mean complex-valued stationary Gaussian random processes. The magnitudes $|c_k(t)|$ are Rayleigh- or Ricean-distributed (dependent on the availability of Line-Of-Sight (LOS) between transmitter and receiver) distributed and the phases Φ (t) are uniformly distributed.

For each weight $\{c_k(t)\}$ there is one stochastic process, characterized by its variance and its power density spectrum $P_k(f)$. The variance is a measure for the average signal power which is received via this path and is defined by the value of ρ_k. $P_k(f)$ determines the average speed of variation in time, i.e. describes the influence of the Doppler effect on the waves arriving at delay time Δ_k. Therefore $P_k(f)$ is also known as Doppler spectrum.

For the description of the channel models the following definitions for the Doppler spectra are used:

A basic parameter is the maximum Doppler frequency.

$$f_d = \frac{v}{\lambda} \tag{B.6}$$

With:

- v the velocity of the receiver or surrounding objects; and
- λ the wavelength of the transmitted signal.

In case that all waves are arriving from all directions at the receiving antenna with approximately the same power the real Doppler spectrum can be approximated by:

$$P_k(f) = \frac{A}{\sqrt{1-\left(\frac{f}{f_d}\right)^2}} \qquad \text{for } f \in \left]-f_d, f_d\right[\tag{B.7}$$

This spectrum is also known as classical Jakes' spectrum and will be denoted as "**Classical**" in the following models.

In the LOS case an additional deterministic component with a distinct Doppler shift has to be added to the Doppler spectrum for the stochastic component. The resultant spectrum denoted as "**Rice**" is defined by the following equation:

$$P_k(f) = \frac{A}{\sqrt{1-\left(\frac{f}{f_d}\right)^2}} + B \times \delta\left(f - f_D\right) \qquad \text{for } f \in \left]-f_d, f_d\right[\tag{B.8}$$

with δ(f) the Dirac pulse and $-f_d \leq f_D \leq f_d$. For a propagation path with a Rice Doppler spectrum the so-called Rice factor is given by B/(π f_d A). It describes the power ratio between the LOS and the stochastic component.

Further spectra are defined with the help of the Gaussian function $G(f, A, f_1, f_2)$:

$$G(f, A, f_1, f_2) = A \exp\left(\frac{\left(f - f_1\right)^2}{2 f_2^2}\right) \tag{B.9}$$

The spectra denoted by "Gauss1" and "Gauss2" consist of a single Gaussian function and are defined as:

$$P_k(f) = G\left(f, A, \pm 0{,}7 \times f_d, 0{,}1 \times f_d\right) \tag{B.10}$$

where the "+" sign is valid for "**Gauss1**" and the "-" sign for "**Gauss2**".

The Gaussian spectra are used in channel profiles for propagation paths with large delay times.

Table B.2: Set of channels

Channel no 7: AWGN			
Velocity: 0 km/h (no time variation)			
Path no, *k*	**Delay (µs)**	**Rel. power (dB)**	**Path type**
1	0,0	0,0	Rice (A = 0, B = 1, f_D = 0 Hz)

Channel no 8: Urban			
Velocities: 2 and 60 km/h (pedestrian and vehicle speed)			
Path no, *k*	**Delay (µs)**	**Rel. power (dB)**	**Path type**
1	0,0	-2,0	Classical
2	0,2	0,0	Classical
3	0,5	-3,0	Classical
4	0,9	-4,0	Classical
5	1,2	-2,0	Classical
6	1,4	0,0	Classical
7	2,0	-3,0	Classical
8	2,4	-5,0	Classical
9	3,0	-10,0	Classical

Channel no 9: Rural			
Velocity: 150 km/h (vehicle speed on highways)			
Path no, *k*	**Delay (µs)**	**Rel. power (dB)**	**Path type**
1	0,0	-4,0	Classical
2	0,3	-8,0	Classical
3	0,5	0,0	Classical
4	0,9	-5,0	Classical
5	1,2	-16,0	Classical
6	1,9	-18,0	Classical
7	2,1	-14,0	Classical
8	2,5	-20,0	Classical
9	3,0	-25,0	Classical

Channel no 10: Terrain obstructed			
Velocity: 60 km/h (speed within built-up areas)			
Path no, *k*	**Delay (µs)**	**Rel. power (dB)**	**Path type**
1	0,0	-8,0	Classical
2	1,0	-2,0	Classical
3	2,5	0,0	Classical
4	3,5	-1,0	Classical
5	5,0	-2,0	Classical
6	8,0	-3,0	Classical
7	12,0	0,0	Classical
8	14,0	-6,0	Classical
9	16,0	-3,0	Classical

Channel no 11: Hilly terrain			
Velocity: 100 km/h (vehicle speed along country roads)			
Path no, *k*	**Delay (µs)**	**Rel. power (dB)**	**Path type**
1	0,0	0,0	Classical
2	0,5	-5,7	Classical
3	1,3	-12,7	Classical
4	1,9	-20,6	Classical
5	30,0	-3,1	Gauss1
6	31,3	-5,4	Gauss1
7	34,9	-11,6	Gauss1
8	37,2	-15,9	Gauss1
9	39,1	-18,9	Gauss1
10	40,0	-25,7	Gauss1
11	80,0	-4,5	Gauss2
12	82,7	-11,5	Gauss2

Channel no 12: SFN			
Velocity: 150 km/h (vehicle speed on highways)			
Path no, *k*	**Delay (µs)**	**Rel. power (dB)**	**Path type**
1	0,0	0,0	Classical
2	100,0	-13,0	Gauss1
3	220,0	-18,0	Gauss2
4	290,0	-22,0	Gauss1
5	385,0	-26,0	Gauss2
6	480,0	-31,0	Gauss1
7	600,0	-32,0	Gauss2

Annex C (informative): Example of mapping of logical frames to multiplex frames

There are many service and stream combinations possible within the DRM system. One example for robustness modes A, B, C and D is illustrated in this annex.

This example DRM signal contains two services: an audio service (service A) and a data service (service D). The audio service also carries a data application.

The audio service uses lower protection. The data application carried with the audio service uses the higher protection. The data service uses the higher protection. The code rates chosen are 0,5 and 0,6 corresponding to protection level 0 and 1 respectively.

Service A consists of two streams: stream 0 carries the audio, stream 1 carries the data application.

Service D consists of one stream: stream 2.

Stream 0 is carried in logical frames L0, stream 1 is carried in logical frames L1 and stream 2 is carried in logical frames L2.

L0 has 0 bytes in the higher protected part (part A) with protection level 0, and 1 105 bytes in the lower protected part (part B) with protection level 1.

L1 has 59 bytes in the higher protected part (part A) with protection level 0.

L2 has 19 bytes in the higher protected part (part A) with protection level 0.

The resulting multiplex frame is illustrated in figure C.1.

Table C.1

Protection level 0		Protection level 1
Stream 1	**Stream 2**	**Stream 0**
59 bytes	19 Bytes	1 105 Bytes

The multiplex description data entity is coded as follows.

Table C.2

Field name	Field size (bits)	Field value
length	7	9
version number	1	0
type	4	0
protection level for part A	2	0
protection level for part B	2	1
data length of part A (stream 0)	12	0
data length of part B (stream 0)	12	1 105
data length of part A (stream 1)	12	59
data length of part B (stream 1)	12	0
data length of part A (stream 2)	12	19
data length of part B (stream 2)	12	0

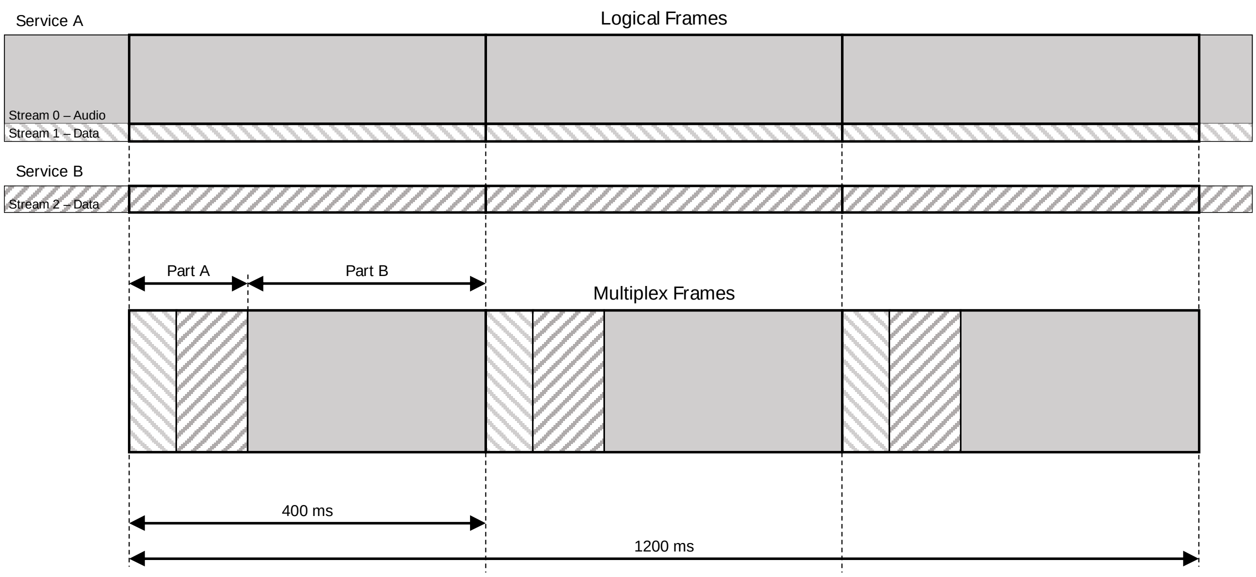

Service A
Logical Frames
Stream 0 – Audio
Stream 1 – Data
Service B
Stream 2 – Data
Part A
Part B
Multiplex Frames
400 ms
1200 ms

Figure C.1

Annex D (normative): Calculation of the CRC word

The implementation of Cyclic Redundancy Check codes (CRC-codes) allows the detection of transmission errors at the receiver side. For this purpose CRC words shall be included in the transmitted data. These CRC words shall be defined by the result of the procedure described in this annex.

A CRC code is defined by a polynomial of degree *n*:

$$G_n(x) = x^n + g_{n-1}x^{n-1} + \ldots + g_2 x^2 + g_1 x + 1$$

with $n \geq 1$:

and: $$g_i \in \{0,1\}, \quad i = 1.....n-1$$

The CRC calculation may be performed by means of a shift register containing *n* register stages, equivalent to the degree of the polynomial (see figure D.1). The stages are denoted by b_0 to b_{n-1}, where b_0 corresponds to 1, b_1 to x, b_2 to x^2, to, b_{n-1} to x^{n-1}. The shift register is tapped by inserting XORs at the input of those stages, where the corresponding coefficients g_i of the polynomial are "1".

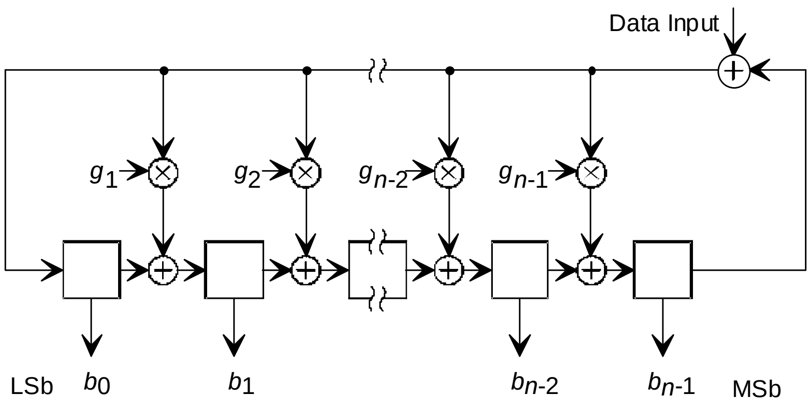


Figure D.1: General CRC block diagram

At the beginning of the CRC calculation, all register stage contents are initialized to all ones. After applying the first bit of the data block (MSb first) to the input, the shift clock causes the register to shift its content by one stage towards the MSb stage (b_{n-1}), while loading the tapped stages with the result of the appropriate XOR operations. The procedure is then repeated for each data bit. Following the shift after applying the last bit (LSb) of the data block to the input, the shift register contains the CRC word which is then read out. Data and CRC word are transmitted with MSb first. The CRC shall be inverted (1's complement) prior to transmission.

The CRC codes used in the DRM system are based on the following polynomials:

- $G_{16}(x) = x^{16} + x^{12} + x^5 + 1$
- $G_8(x) = x^8 + x^4 + x^3 + x^2 + 1$
- $G_6(x) = x^6 + x^5 + x^3 + x^2 + x + 1$
- $G_5(x) = x^5 + x^4 + x^2 + x + 1$
- $G_3(x) = x^3 + x + 1$
- $G_2(x) = x^2 + x + 1$

- $G_1(x) = x + 1$

The assignment of the polynomials to the respective applications is given in each clause.

Annex E (informative): RF protection ratios

The combinations of spectrum occupancy types and robustness modes of DRM signals lead to several transmitter RF spectra, which cause different interference and therefore require different RF protection ratios, see Recommendation ITU-R BS.1615 [i.1] and Recommendation ITU-R BS.1660 [i.2].

Annex F (informative): Alternative Frequency and announcement signalling

F.0 Introduction

The DRM system can signal alternative frequencies for the whole DRM multiplex or some DRM services of the tuned DRM multiplex to allow the receiver to counter reception problems by automatically and quickly switching to an alternative frequency providing better reception conditions.

It is also possible to signal service linking information which allows the service provider to establish one or more sets of identifiers that carry identical, in the case of a hard link, or related, in the case of a soft link, content. The set of identifiers is called a linkage set. There may be several linkage sets that are valid at different times of day. Each linkage set is identified by the Linkage Set Number together with a set of flags, and by use of the Linkage Actuator, linkage sets can be activated and deactivated. The receiver uses these linkage sets during service selection and service following to determine a set of candidate services, potentially on different bearers, that are equivalent or related to the selected service. The receiver selects an appropriate service from these candidates based on criteria such as service availability and quality.

In addition, the DRM system can signal announcements (e.g. traffic or news announcements), so that the receiver can automatically switch to another DRM service or even to another broadcast system for the duration of an active announcement.

These features are signalled using a combination of SDC data entities. This annex describes the general capabilities of the alternative frequency signalling, linking and announcement features. It also explains how the different SDC entities work together to enable this kind of signalling.

F.1 Possibilities of the Alternative Frequency Signalling feature

Using the Alternative Frequency feature, the broadcaster can signal to the receiver alternative frequencies for the following items:

- the whole DRM multiplex, being broadcast identically and synchronously on other frequencies; the receiver can check whether it receives the identical DRM multiplex on the indicated frequencies and also their reception quality; if required the receiver can instantly switch to another frequency without service interruption;
- the whole DRM multiplex (all services with the same Service identifiers), but with different channel parameters and/or multiplex timing (non-synchronous); frequency switching causes a service interruption;
- single services of the tuned DRM multiplex; frequency switching causes a service interruption; single services can be available:
    - in other DRM multiplexes using the same Service identifiers;
    - in other broadcast systems (e.g. AM, FM, FM-RDS, DAB) or other DRM multiplexes using a different Service identifier;
- the frequency of the enhancement layer from the base layer, or vice versa.

The validity of alternative frequency lists can be restricted to certain times (schedule definition feature) and/or to certain geographic areas (region definition feature). The schedule definition feature is based on a weekly schedule. The region definition feature allows the definition of geographic areas by longitude/latitude plus extent as well as by internationally standardized CIRAF zones.

If the list of alternative frequencies links to a service carried using another broadcast system, or to a service with a different service identifier, it can be indicated whether the other service carries the identical audio programme or a similar one. The receiver will try to switch to the "same service" before trying to switch to an alternative service. If it is necessary to signal changes to these equivalent or related services dynamically and/or for short durations then service linking may be used additionally to alternate frequencies in order to build up linkage sets and then rapidly switch between them.

DRM receivers should store the complete alternative frequency information (all five data entity types 3, 4, 7, 11 and 13) when assigning a DRM service to a station button (see clause G.2). Therefore it is sensible for the broadcaster to signal his complete frequency schedule to the receiver. The frequency schedule should provide the full week's changes rather than only provide a subset, for example only indicating daytime frequencies during daytime transmissions and nighttimes frequencies during night time transmissions. This permits faster start-up of a service if the frequency of the DRM multiplex is different from the last time the service was selected.

It is essential that all frequencies used in a synchronous multi-frequency network are signalled because the bitstream from all transmitters in the network are identical. This is also highly recommended even if the network is not synchronous because receivers can then store all the possible frequencies when defining a station button including the principle tuned frequency of the multiplex. This frequency information is necessary if the receiver has to tune away and then wants to get back again. It is also required to help a receiver to identify its current region (see clause F.3).

It is recommended that broadcasters specify all alternative sources of each service and include all frequencies. Receivers will sequentially test all given frequencies. If no frequency information is provided, then receivers have to scan for the service identifier if they are to find the service, and that may take so long as to provide an unacceptable user experience.

Broadcasters should note that links to completely analogue broadcast systems have unpredictable results for the receiver, because receivers are unable to check that the correct service is received on the given frequency.

If the broadcaster uses scheduled frequency information he will provide SDC data entity "Time and date information data entity - type 8" as the time reference for the schedules in "Alternative frequency signalling: Schedule definition data entity - type 4".

F.2 Possibilities of the announcement feature

The announcement feature can be used to interrupt the currently presented audio programme by another providing short clips of information.

Using the announcement feature, the broadcaster can signal to a receiver:

- which types of announcements are provided;
- which type of announcement is currently active;
- whether the announcement content is carried by a DRM service within the current DRM multiplex or by a service on another frequency or from a different broadcast system like FM or DAB.

This information can be specified for each DRM service or for any combination of DRM services within the tuned DRM multiplex.

If the announcement signalling directs the receiver away from the tuned multiplex to another service carrying the announcement content (e.g. on a different type of broadcast system) then the other service will provide the mechanism to indicate the end of the announcement such that the original listening can be restored. The service identifiers for the other services carrying the announcement content (and optionally their frequencies) are signalled in the SDC by the "Alternative frequency signalling: Other services data entity - type 11" with the "Short Id/Announcement Id flag" set to 1.

EXAMPLE: Within the tuned DRM multiplex the broadcaster provides four DRM services named A, B, C and D. A and B provide English programmes while C and D carry German versions of the programmes.

The broadcaster may use the announcement feature to signal to the receiver that in case of a traffic announcement service A should switch to service B, while service C should switch to service D. Alarm information may only be available in English, so services A, C and D should switch to service B in case of available warning information. The news channel is not broadcast in the tuned DRM multiplex, so service A switches to a DAB service (specifying the other service identifier along with an optional list of frequencies) while service C switches to an FM-RDS service.

F.3 SDC data entities overview for Alternative Frequency and announcement signalling

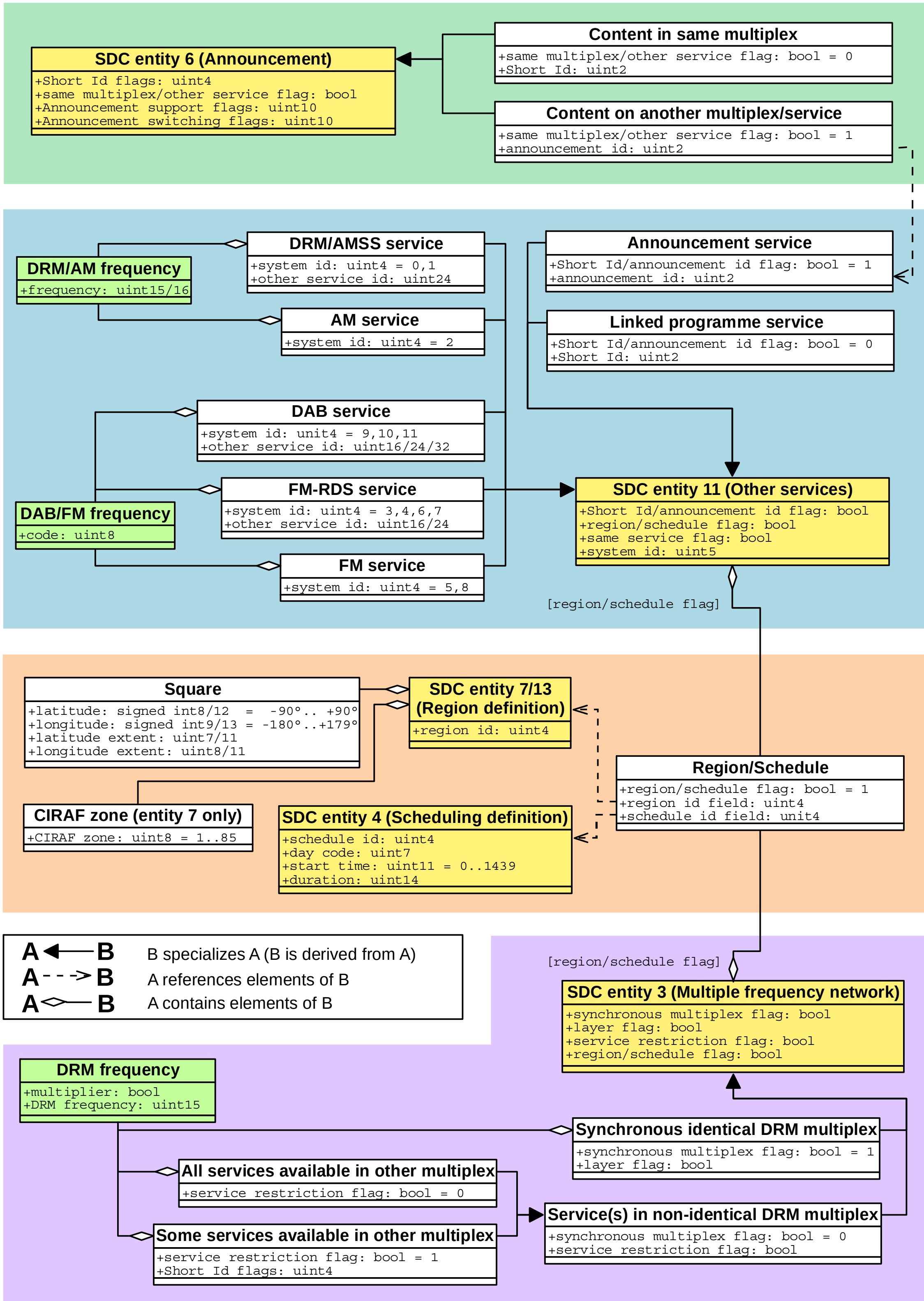


Figure F.1: Alternative Frequency and Announcement Signalling - Involved SDC data entities

F.4 SDC data entities and setup for alternative frequency signalling

The following SDC entities are used to carry the alternative frequency and announcement signalling:

- Alternative frequency signalling: Schedule definition data entity - type 4.
- Alternative frequency signalling: Region definition data entity - type 7.
- Alternative frequency signalling: detailed region definition data entity - type 13.
- Alternative frequency signalling: Multiple frequency network information data entity - type 3.
- Alternative frequency signalling: Other services data entity - type 11.

To set up the alternative frequency signalling feature, the broadcaster will typically perform the following steps:

1) If some alternative frequencies are only valid at certain times, provide up to 15 schedules (SDC data entities type 4), each identified by its unique Schedule Id; to explicitly indicate that some frequencies are available all the time, it is recommended that a "24 hours, 7 days a week" schedule is signalled.

2) If some alternative frequencies are only valid in certain geographic regions, provide up to 15 region definitions (SDC data entities type 7 and 13), each identified by its unique Region Id.

3) If alternative frequencies are available for the current DRM multiplex or at least some services, provide SDC data entities type 3 for all lists of frequencies; each list may point to alternative frequencies carrying the identical DRM multiplex in a synchronous way (seamless alternative frequency checking and switching may be performed by the receiver) or carrying some or all services of the current DRM multiplex with different channel parameters and/or not synchronized (checking and switching will interrupt the service presentation); each list of frequencies may be restricted to a geographic area and/or a schedule by referencing one SDC data entity 4 and/or 7 and/or 13.

4) If alternative frequencies are available for individual services of the current DRM multiplex using different DRM Service identifiers or being carried on a different broadcast system (e.g. DAB or FM-RDS), provide SDC data entities type 11 for all lists of frequencies; each list can indicate one other service identifier, the broadcast system type and the "same service" flag along with a list of frequency values;
if only a service identifier is specified (without any frequencies), the receiver has to scan for an available frequency; each list of frequencies may be restricted to a geographic area and/or a schedule by referencing one SDC data entity 4 and/or 7 and/or 13.

A broadcaster can choose to group the frequencies by region by using a common region id per group even if no "Alternative frequency signalling: Region definition data entity - type 7" or "Alternative frequency signalling: detailed region definition data entity - type 13" is provided for a particular Region Id. This allows receivers to check alternative frequencies within the same group first (defined by the Region Id) before checking other groups - the current position of the receiver does not need to be known.

EXAMPLE: Broadcaster X broadcasts a service on frequencies 6 200 kHz and 9 500 kHz in Europe and on frequencies 11 600 kHz and 13 800 kHz in Africa. These four frequencies should be sent as two groups of two using different Region Ids for Europe and Africa, even if no Region definition (data entity type 7 or 13) information is given.

F.5 SDC data entities and setup for announcement

The following SDC entities are used to carry the Alternative Frequency and Announcement signalling:

- Alternative frequency signalling: Schedule definition data entity - type 4.
- Alternative frequency signalling: Region definition data entity - type 7.
- Alternative frequency signalling: detailed region definition data entity - type 13.

- Alternative frequency signalling: Other services data entity - type 11.
- Announcement support and switching data entity - type 6.

To set up the announcement signalling, the broadcaster will perform the following steps:

1) Set up all required SDC data entities type 6, indicating which types of announcements are provided and which are currently active, and which internal DRM services are linked to which services carrying the announcement content (services in the same DRM multiplex or other services).

2) If some alternative frequencies for announcements are only valid at certain times, provide up to 15 schedules (SDC data entities type 4), each identified by its unique Schedule Id; to explicitly indicate that some frequencies are available all the time, it is recommended that a "24 hours, 7 days a week" schedule is signalled.

3) If some alternative frequencies for announcements are only valid in certain geographic regions, provide up to 15 region definitions (SDC data entities type 7 and 13), each identified by its unique Region Id.

4) For every "Announcement Id" value provided by a SDC data entities type 6 (thereby linking to another service), there should be at least one SDC data entity type 11 (with the "Short Id/Announcement Id flag" being set to 1 and using the same "Announcement Id"), providing the broadcast system type, service identifier and frequencies of the other service.

F.6 Alternative frequency and announcement signalling - coding example

Situation

Broadcaster A transmits to the UK and to North America excluding the US.

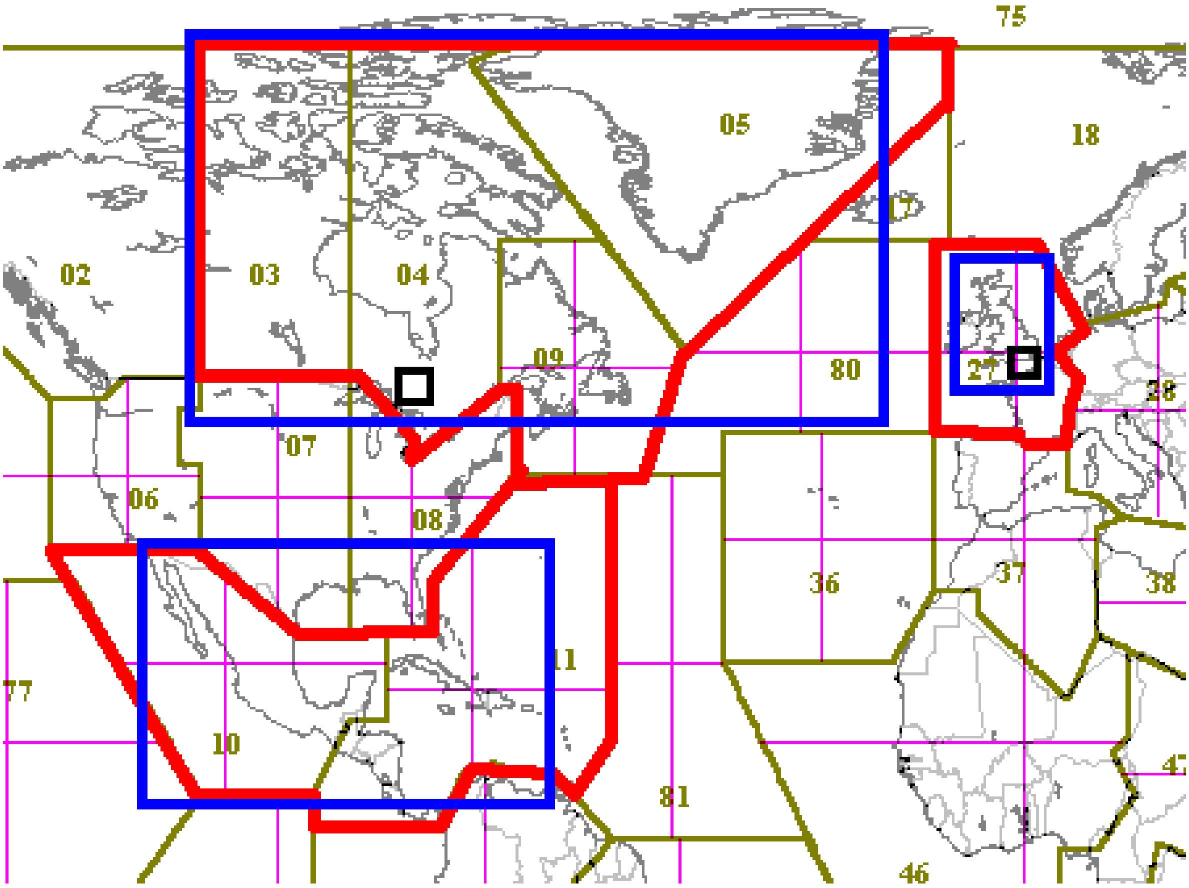


Figure F.2: Coding Example for Alternative Frequency and Announcement Signalling - regions and CIRAF zones

The service is available:

1) On SW freq. 1, DRM daily from 16:00 to 02:00 UTC using Service identifier 1 (all marked regions).

2) On SW freq. 2, DRM daily from 12:00 to 18:00 UTC using Service identifier 1 only Canada, Greenland, UK. This DRM multiplex is synchronous to the first one.

3) On SW freq. 3, AM daily from 16:00 to 02:00 UTC (no Service identifier, all marked region).

4) On MW DRM, UK, daily, 24 hours.

5) On FM in several parts of UK, using many different frequencies, normally daily, 24 hours, but sometimes there is a different programme on this network (e.g. sports coverage) but a different PI code is then being used.

6) On FM in London on 2 different frequencies, daily, 24 hours.

7) On FM in a City in North America, weekdays from 18:00 to 22:00 UTC, weekend from 12:00 to 16:00 UTC.

8) On DAB, daily, 24 hours, in the UK only.

9) In the UK, traffic messages can be obtained from a different service on DAB, daily, 24 hours a day.

10) In North America the receiver should tune to the DRM transmission of Broadcaster B when no transmission is available.

Encoded SDC Entities

Schedules (SDC entities type 4)

- Schedule Id 1 | Day Code = 1234567 | Start = 00:00 UTC| Duration = 24 h
  alternative coding for the same information:
  Schedule Id 1 | Day Code = 1 | Start = 00:00 UTC| Duration = 7×24 h
  (means: 24 h, daily).
- Schedule Id 2 | Day Code = 1234567 | Start = 16:00 UTC | Duration = 10 h.
- Schedule Id 3 | Day Code = 1234567 | Start = 12:00 UTC | Duration = 6 h.
- Schedule Id 4 | Day Code = 12345 | Start = 18:00 UTC | Duration = 4 h.
- Schedule Id 4 | Day Code = 67 | Start = 12:00 UTC | Duration = 4 h
  (means 18:00 to 22:00 UTC on weekdays and 12:00 to 16:00 UTC on Saturday and Sunday).

Regions (SDC entities type 7)

- Region Id 1 | upper blue rectangle North America | CIRAF 3, 4, 5, 9, 27 (red)
  Region Id 1 | lower blue rectangle North America | CIRAF 10, 11 (red)
  (means: all marked regions).
- Region Id 2 | upper blue rectangle North America | CIRAF 3, 4, 5, 9, 27 (red).
- Region Id 3 | blue rectangle UK.
- Region Id 4 | black rectangle London.
- Region Id 5 | black rectangle North American City.

DRM Services/Frequencies - same Service identifiers (SDC entities type 3)

- Synchronous Multiplex flag = 1 | Layer flag = 0 | Region Id 1 | Schedule Id 2 | SW-Freq. 1
  (see number 1 above).
- Synchronous Multiplex flag = 1 | Layer flag = 0 | Region Id 2 | Schedule Id 3 | SW-Freq. 2
  (see number 2 above).

Other Services/Frequencies - different service identifiers (SDC entities type 11)

- Same Service flag = 1 | System Id = 00010 (AM without Id) | Region Id 1 | Schedule Id 2 | SW Freq 3
  (see number 3 above).
- Same Service flag = 1 | System Id = 00010 (AM without Id) | Region Id 3 | Schedule Id 1 | MW Freq
  (see number 4 above).
- Same Service flag = 1 | System Id = 00011 (FM-RDS with ECC, Europe and North America) | Region Id 3 | Schedule Id 0 | ECC + PI 1
  (see number 5 above).
- Same Service flag = 1 | System Id = 00100 (FM-RDS without ECC, Europe and North America) | Region Id 4 | Schedule Id 1 | PI 2 | FM Freq. 1, FM Freq. 2
  (see number 6 above).
- Same Service flag = 1 | System Id = 00101 (FM without RDS, Europe and North America) | Region Id 5 | Schedule Id 1 | FM Freq. 3
  (see number 7 above).

- Same Service flag = 1 | System Id = 01001 (DAB with ECC + SId) | Region Id 3 | Schedule Id 1 | ECC + SId 1 | DAB Freq. 1
  (see number 8 above).

- Same Service flag = 0 | System Id = 00000 (DRM) | Region Id 1 | Schedule Id 0 | DRM Service identifier of Broadcaster B | DRM Freqs 1-n
  (see number 10 above).

Announcements (SDC entities type 6)

- Short Id 1 | Other Service flag = 1 | Announcement Id = 1 | announcement support flags | announcement switching flags.

Other Service/Frequencies - for Announcement (SDC entities type 11)

- Announcement Id flag = 1 | Announcement Id 1 | System Id = 01001 (DAB with ECC + SId) | Region Id 3 | Schedule Id 1 | ECC + SId 2 | DAB Freq. 1
  (see number 9 above).

Annex G (informative): Guidelines for receiver implementation

G.0 Introduction

This annex provides some guidelines for receiver behaviour. It does not imply that all types of receivers will include all the described features.

G.1 Alternative Frequency checking and Switching (AFS)

If the receiver notices that the currently selected DRM service has reception problems, it should check for alternative frequencies in the following way:

1) First the receiver tries to find an alternative frequency that provides the identical DRM multiplex as the currently tuned frequency. The receiver checks for DRM multiplexes that are identical to the currently tuned multiplex and synchronous to it. This information is available in SDC data entities "Alternative frequency signalling: Multiple frequency network information data entity - type 3"; all suitable DRM multiplexes have the "Synchronous Multiplex flag" set to 1.
   For identical and synchronous DRM multiplexes the receiver can check for the availability of the identical DRM multiplex on another frequency (and switch) without service interruption, see clause G.3.

2) If no identical and synchronous DRM multiplex is available the receiver checks other DRM frequencies that carry at least the currently selected DRM service(s). On a non-identical DRM multiplex the channel parameter and the service structure can be different (e.g. more or less services; with or without multimedia data) and the currently selected service might be carried with different parameters (e.g. different bit rate or audio mode). The SDC data entities "Alternative frequency signalling: Multiple frequency network information data entity - type 3" ("Synchronous Multiplex flag" has the value 0) lists all alternative DRM multiplexes that carry one or more DRM services of the current DRM multiplex.

   If the receiver can present multiple services at the same time (i.e. audio and multimedia) the receiver should first try the alternative frequencies that carry all currently selected services.

   Caution: checking for the availability of a DRM service on another frequency causes a service interruption (regardless whether the DRM service is available on another frequency or not).

3) If "Alternative frequency signalling: Other services - data entity type 11" (marked as "same service") are available that signal at least one alternative frequency, the receiver checks these frequencies. If none of the given frequencies is valid then the receiver checks all given frequencies for other services NOT marked as "same service".

4) If "Alternative frequency signalling: Other services - data entity type 11" signals a DRM Service identifier or another service identifier but no frequency, then the receiver has to scan for the selected DRM Service identifier or for the other service identifier. It starts scanning for "same services" before trying NOT "same services".

   Scanning will usually not be sensible if the selected DRM service signals that the alternative source is currently not broadcast (described by scheduled frequency lists) or that it is not receivable in the receiver's region (indicated by region definitions).
   If no alternative frequencies are signalled at all or no valid alternative frequency could be found then the receiver could scan for the current DRM service and any known alternative other services. Such a scanning can take a very long time!

NOTE 1: In step 4 the broadcaster explicitly asks the receiver to scan for a given other service identifier (by not providing any frequency). In step 5 the receiver assumes that the alternative frequency information provided by the broadcaster is not complete.

5) If also step 5 fails then there is no alternative source for the current DRM service available.

When the receiver checks for alternative frequencies (steps 1 to 4) it will also take into account the SDC data entities "Alternative frequency signalling: Region definition data entity - type 7", "Alternative frequency signalling: detailed region definition data entity - type 13" and "Alternative frequency signalling: Schedule definition data entity - type 4" to determine when and where a certain alternative frequency is valid. Note that SDC data entity "Time and date information data entity - type 8" provides the time reference for the schedules in "Alternative frequency signalling: Schedule definition data entity - type 4".

If a receiver does not know its current position or if the "Alternative frequency signalling: Region definition data entity - type 7" or "Alternative frequency signalling: detailed region definition data entity - type 13" describing the region has not yet been received (or is not broadcast), it might still evaluate the Region Id. The receiver can determine the Region Id of the currently tuned frequency and thus first check the alternative frequencies belonging to the same region.

NOTE 2: These proposed steps do not tell when a receiver should switch to an alternative frequency nor do they forbid that the receiver selects DAB services before DRM ones. But the receiver should never try NOT-"same services" unless all other sources of the selected DRM service have failed. NOT-"same services" are the last resort!

NOTE 3: Probably a receiver will try seamless AFS ("Seamless Alternative Frequency Checking and Switching"; see clause G.3) to identical and synchronous DRM multiplexes even while the currently tuned frequency has no reception problems. In case the reception of the currently tuned frequency fails the receiver can then find an alternative frequency faster.

G.2 Station buttons for DRM services

If the user assigns a DRM service to a station button it is recommended for the DRM receiver to store all alternative frequency information (all four data entity types 3, 4, 7, 11 and 13) as well as the currently used frequency and the DRM Service identifier. This permits the receiver to find the right frequency even if the service is selected (by the user pressing the station button) while the service is broadcast on another frequency compared to the time the station button was defined.

If such a station button is selected the expected behaviour of the receiver would be as follows:

a) Check on what frequency the service was received the last time. If this frequency was marked "same service" in the alternative frequency information the receiver should try tuning to this frequency. If tuning succeeds (the desired DRM Service identifier is available on the frequency) tuning is successfully finished. If the receiver did not find the expected service identifier it has to start with step 1 of the above list (see clause G.1).

b) If the frequency that was successfully tuned to the last time is NOT marked as "same service" then the receiver directly starts with step 1 of the above list (see clause G.1).

c) If the desired DRM service cannot be found the receiver might then offer the user to scan for this DRM service, i.e. try to find the DRM Service identifier on another frequency.
Note that scanning might be sensible if the receiver did not switch to a certain DRM service for a long time (i.e. if the frequencies might have changed in the meantime). Scanning will usually not be sensible if the user tries to tune to a DRM service that is broadcast just part of the day (described by scheduled frequency lists) and that provides no frequencies for the current day of week/time.

The "same service" flag permits faster start of service presentation and by setting this flag to "0" the broadcaster can order the receiver to first try "same services" before NOT-"same services".

Note that for scheduled frequencies it is useful to have a battery powered real time clock that provides at least day of week and time (UTC) when the receiver is switched on.

G.3 Seamless Alternative Frequency checking and Switching (AFS)

Seamless Alternative Frequency Switching (AFS) provides the functionality of seamlessly checking for the availability of the same programme material on a differing frequency and then switching to it if it is valid. Alternative frequencies can be signalled by use of SDC data entity type 3. The various steps of this process are indicated in figure G.1.

AFS specific mathematical symbols are defined as follows:

T_d: time delay at point of reception between the current and the possible alternative frequency.

T_{tune}: time needed by the receiver to tune to the alternative frequency.

T_{check}: time available to acquire the data required for the validation of the AF.

Procedure:

At the start of known SDC block on the tuned frequency, the receiver re-tunes to the alternative frequency. It acquires the data necessary to perform the AF-check and immediately tunes back to the original tuned frequency. This process has to be completed within the time interval T_{check}. Subsequently the validity of the alternative frequency can be computed before the next occurrence of the SDC. Subject to the validation of the alternative frequency, the receiver may choose to switch to the new frequency at this point without an interruption of service.

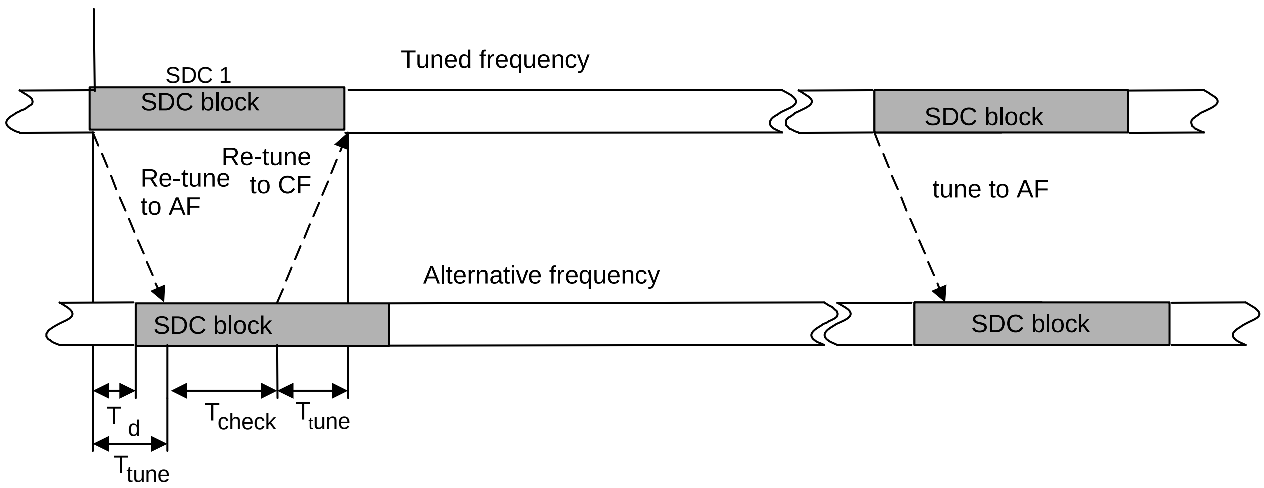


Figure G.1: Illustration of AFS function

The points at which the receiver may check the alternative frequencies are governed by the Identity field in the FAC in combination with the AFS index signalled in the SDC.

If the receiver detects a failure of the FAC CRC for the first transmission frame of the transmission super frame then it cannot perform an AFS check because the value of the Identity field is unknown.

For fully dynamic operation (see clause 6.4.5), no AFS is possible because the receiver has no knowledge of the data that will be sent in future SDC blocks.

For fully static operation, the AFS function may be performed every super transmission frame, provided that the receiver has stored all the different SDC blocks in the cycle. The number of SDC blocks in the cycle is given by the AFS index + 1.

For semi-dynamic operation, the AFS function may only be performed at certain transmission super frames. The following examples illustrate some of the many possibilities.

EXAMPLE 1: Changing the content of the SDC block (A to B) with AFS index = 0.

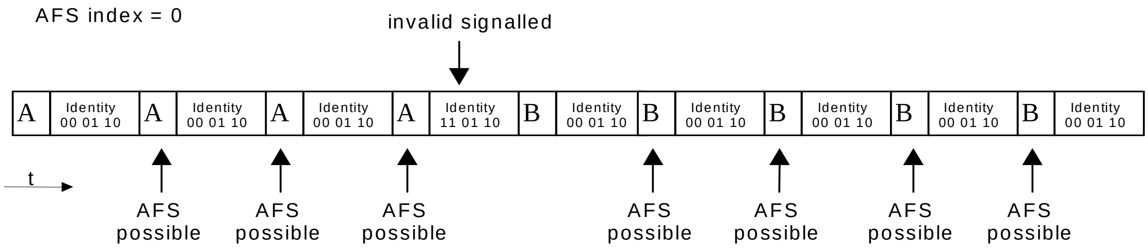


Figure G.2: Example 1

NOTE 1: Very fast AFS possibility after tuning;
Very limited SDC data size when AFS feature should be used.

EXAMPLE 2: Changing the content of both SDC blocks (A to C; B to D) with AFS index = 1.

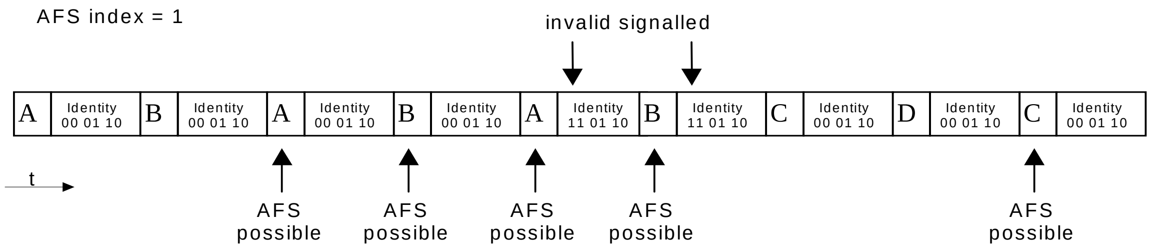


Figure G.3: Example 2

NOTE 2: While changing the SDC blocks in two consecutive SDC blocks is no AFS possible;
With AFS = n the first AFS can occur after (n + 1) received SDC blocks.

EXAMPLE 3: Changing the content of one SDC block (A to C) with AFS index = 1.

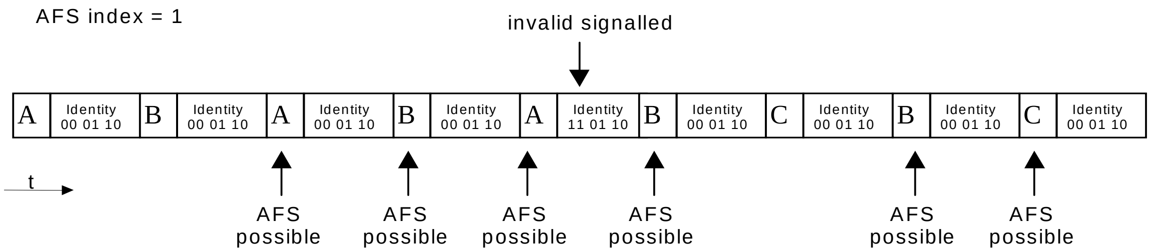


Figure G.4: Example 3

NOTE 3: Only one AFS possibility is missed.

EXAMPLE 4: Continuous changing of one SDC block (B to C to D.) with AFS index = 1.

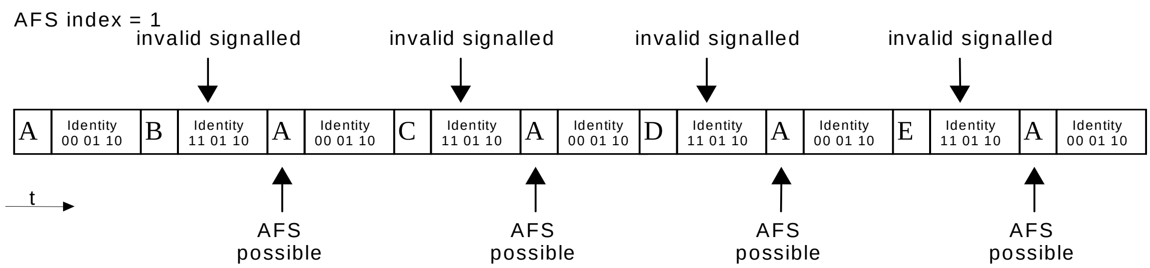


Figure G.5: Example 4

NOTE 4: Only every second frame AFS is possible;
SDC data size is increased.

EXAMPLE 5: Change of repetition rate of SDC block (without reconfiguration) with AFS index = 1.

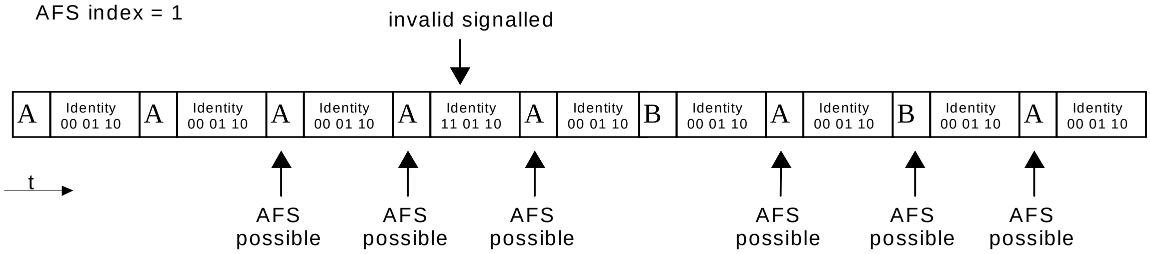


Figure G.6: Example 5

NOTE 5: First AFS possible after n frames after tuning;
Flexible SDC data size.

For other values of AFS index, similar schemes can be applied.

Annex H (informative): Service capacity and bit rates

The orders of magnitude of the available total bit rates, which depend upon the signal bandwidth, the protection mode, and the error correction code rates are given below.

For robustness modes A, B C and D tables H.1 and H.2 apply.

Table H.1: 64-QAM modulation, coding rate of 0,6 for the MSC (EEP)

Robustness mode	Spectrum occupancy					
	0	**1**	**2**	**3**	**4**	**5**
A	11,3 kbit/s	12,8 kbit/s	23,6 kbit/s	26,6 kbit/s	49,1 kbit/s	55 kbit/s
B	8,7 kbit/s	10 kbit/s	18,4 kbit/s	21 kbit/s	38,2 kbit/s	43 kbit/s
C	-	-	-	16,6 kbit/s	-	34,8 kbit/s
D	-	-	-	11 kbit/s	-	23,4 kbit/s

Table H.2: 16-QAM modulation, coding rate of 0,62 for the MSC (EEP)

Robustness mode	Spectrum occupancy					
	0	**1**	**2**	**3**	**4**	**5**
A	7,8 kbit/s	8,9 kbit/s	16,4 kbit/s	18,5 kbit/s	34,1 kbit/s	38,2 kbit/s
B	6 kbit/s	6,9 kbit/s	12,8 kbit/s	14,6 kbit/s	26,5 kbit/s	29,8 kbit/s
C	-	-	-	11,5 kbit/s	-	24,1 kbit/s
D	-	-	-	7,6 kbit/s	-	16,3 kbit/s

Minimum absolute (R = 0,50, 16-QAM, robustness mode B, 4,5 kHz) 4,8 kbit/s.

Maximum absolute (R = 0,78, 64-QAM, robustness mode A, 20 kHz) 72 kbit/s.

For robustness mode E and spectrum occupancy 0, the table H.3 applies for the MSC (EEP):

Table H.3

Constellation	Code rate	Bit rate
4-QAM	0,25	37,3 kbit/s
4-QAM	0,333	49,7 kbit/s
4-QAM	0,4	59,6 kbit/s
4-QAM	0,5	74,5 kbit/s
16-QAM	0,33	99,4 kbit/s
16-QAM	0,411	122,4 kbit/s
16-QAM	0,5	149,1 kbit/s
16-QAM	0,625	186,4 kbit/s

Annex I:
Void

Annex J (informative): Numbers of input bits

Table J.1: Number of input bits L per multiplex frame for EEP robustness mode A

Parameters	Spectrum occupancy					
	0	1	2	3	4	5
64-QAM, R_{all} = 0,5	3 757	4 248	7 878	8 857	16 374	18 336
64-QAM, R_{all} = 0,6	4 509	5 096	9 450	10 628	19 646	21 998
64-QAM, R_{all} = 0,71	5 322	6 018	11 157	12 547	23 193	25 976
64-QAM, R_{all} = 0,78	5 898	6 664	12 364	13 908	25 704	28 788
16-QAM, R_{all} = 0,5	2 505	2 832	5 250	5 904	10 914	12 222
16-QAM, R_{all} = 0,62	3 131	3 540	6 565	7 381	13 645	15 280

Table J.2: Number of input bits L per multiplex frame for EEP robustness mode B

Parameters	Spectrum occupancy					
	0	1	2	3	4	5
64-QAM, R_{all} = 0,5	2 880	3 312	6 133	6 991	12 727	14 304
64-QAM, R_{all} = 0,6	3 456	3 972	7 361	8 390	15 272	17 162
64-QAM, R_{all} = 0,71	4 080	4 692	8 688	9 900	18 026	20 264
64-QAM, R_{all} = 0,78	4 520	5 196	9 630	10 980	19 980	22 456
16-QAM, R_{all} = 0,5	1 920	2 208	4 089	4 662	8 484	9 534
16-QAM, R_{all} = 0,62	2 400	2 760	5 111	5 826	10 606	11 920

Table J.3: Number of input bits L per multiplex frame for EEP robustness mode C

Parameters	Spectrum occupancy					
	0	1	2	3	4	5
64-QAM, R_{all} = 0,5	Not used			5 514	Not used	11 581
64-QAM, R_{all} = 0,6				6 615		13 898
64-QAM, R_{all} = 0,71				7 808		16 406
64-QAM, R_{all} = 0,78				8 654		18 188
16-QAM, R_{all} = 0,5				3 675		7 722
16-QAM, R_{all} = 0,62				4 595		9 651

Table J.4: Number of input bits L per multiplex frame for EEP robustness mode D

Parameters	Spectrum occupancy					
	0	1	2	3	4	5
64-QAM, R_{all} = 0,5	Not used			3 660	Not used	7 800
64-QAM, R_{all} = 0,6				4 391		9 359
64-QAM, R_{all} = 0,71				5 185		11 050
64-QAM, R_{all} = 0,78				5 746		12 242
16-QAM, R_{all} = 0,5				2 439		5 199
16-QAM, R_{all} = 0,62				3 050		6 500

Table J.5: Number of input bits L per multiplex frame for EEP robustness mode E

Parameters	Spectrum occupancy					
	0	1	2	3	4	5
16-QAM, R_{all} = 0,33	9 938	Not used				
16-QAM, R_{all} = 0,41	12 243					
16-QAM, R_{all} = 0,5	14 907					
16-QAM R_{all} = 0,62	18 635					
4-QAM, R_{all} = 0,25	3 727					
4-QAM, R_{all} = 0,33	4 969					
4-QAM, R_{all} = 0,4	5 962					
4-QAM, R_{all} = 0,5	7 454					

Table J.6: Number of input bits L per SDC block for robustness mode A

Parameters	Spectrum occupancy					
	0	1	2	3	4	5
16-QAM, R_{all} = 0,5	321	366	705	798	1 494	1 680
4-QAM, R_{all} = 0,5	161	184	353	399	748	840

Table J.7: Number of input bits L per SDC block for robustness mode B

Parameters	Spectrum occupancy					
	0	1	2	3	4	5
16-QAM, R_{all} = 0,5	246	288	552	630	1 164	1 311
4-QAM, R_{all} = 0,5	124	144	276	316	582	656

Table J.8: Number of input bits L per SDC block for robustness mode C

Parameters	Spectrum occupancy					
	0	1	2	3	4	5
16-QAM, R_{all} = 0,5	Not used			564	Not used	1 200
4-QAM, R_{all} = 0,5				282		601

Table J.9: Number of input bits L per SDC block for robustness mode D

Parameters	Spectrum occupancy					
	0	1	2	3	4	5
16-QAM, R_{all} = 0,5	Not used			291	Not used	651
4-QAM, R_{all} = 0,5				146		326

Table J.10: Number of input bits L per SDC block for robustness mode E

Parameters	Spectrum occupancy					
	0	1	2	3	4	5
4-QAM, R_{all} = 0,5	930	Not used				
4-QAM, R_{all} = 0,25	465					

Annex K (informative): Simulcast transmission, alternate sources and enhancement signalling

The DRM signal is designed to work in the same broadcast bands as analogue signals. The DRM system can cross-refer to the same or related services carried in another DRM signal, or in signals using AM, FM or DAB systems. Future quality enhancement is also permitted by the signalling to allow a second DRM multiplex to provide additional data capacity for services. See clause 6 for the detailed explanation of how this information is provided. This facility can be used in many ways, with some examples given in figure K.1.

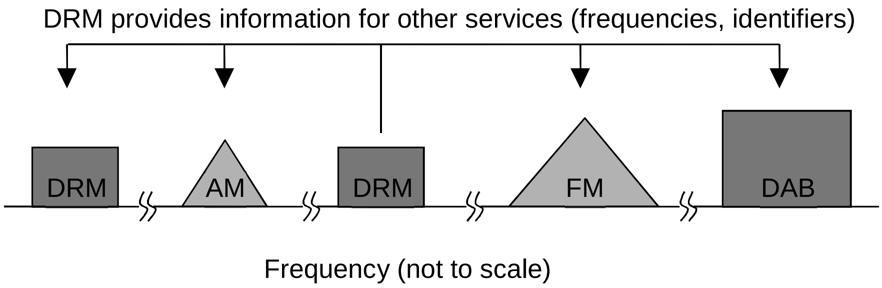


Figure K.1: Example of linking to related services

Simulcast transmission of services using DRM and AM can be performed by the juxtaposition of the analogue AM signal and a DRM digital signal.

Figures K.2 and K.3 illustrate some solutions for transmitting the AM and DRM signals from a single transmitter. They can equally be produced by two separate transmitters.

Figure K.2 gives some possibilities for the case where the DRM reference frequency, f_R, is one channel or two channels (i.e. ±9 kHz, ±10 kHz, -18 kHz or -20 kHz) from the AM carrier frequency, f_C, and figure K.3 gives some possibilities for the case where the DRM reference frequency, f_R, is nominally half a channel from the AM carrier frequency, f_C. Due to the requirement to position the DRM reference frequency on an integer multiple of 1 kHz, the DRM reference frequency and the AM carrier frequency will be either 4 kHz or 5 kHz apart.

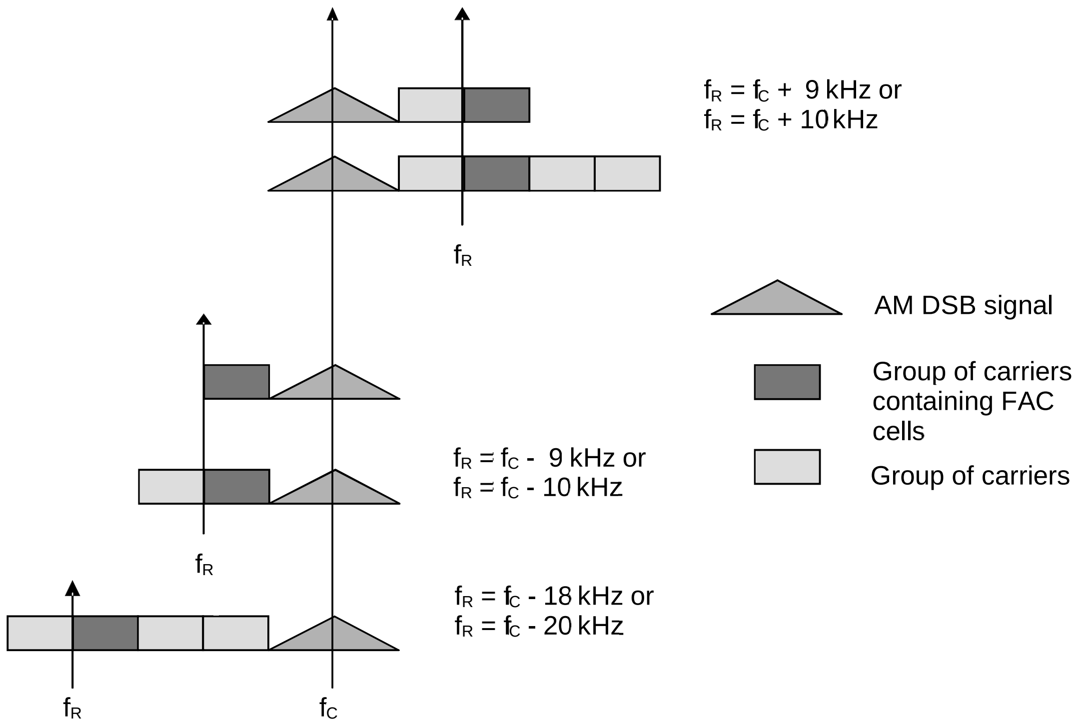


Figure K.2: Example simulcast modes for whole channel offsets

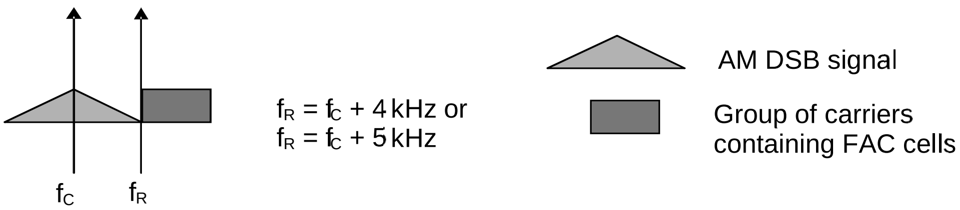


Figure K.3: Example simulcast modes for half channel offsets

Figures K.4 to K.6 illustrate some examples of the use of the base/enhancement signalling of DRM to provide solutions for transmitting higher quality DRM or DRM and AM signals from a single transmitter.

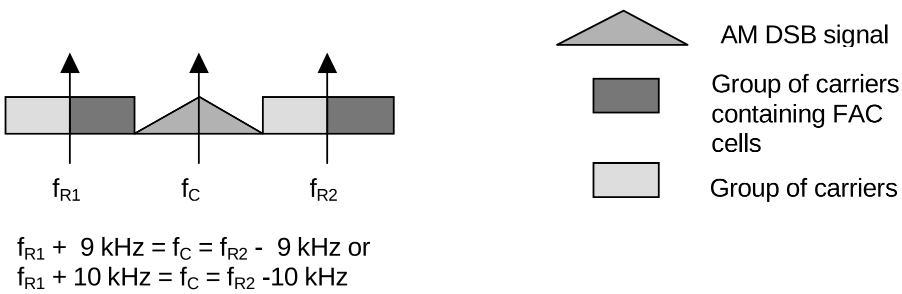


Figure K.4: Example simulcast modes with two DRM signals

In figure K.4, the two DRM signals may be alternate frequencies for the same multiplex, providing spectral diversity. In this case, the AFS data entities (see clause 6) indicate the two frequencies that the DRM multiplex is available on. Alternatively, the two DRM signals may be the base layer and enhancement layer for a multiplex. In this case the AFS data entities signal the frequency of the other layer. All receivers will be able to decode the base layer, whilst some will also be able to decode the enhancement layer and provide higher quality.

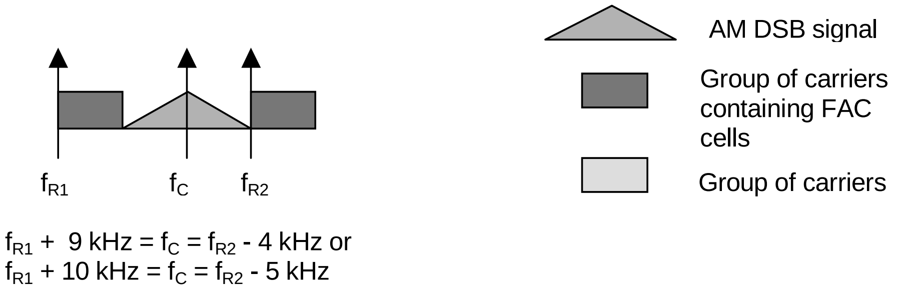


Figure K.5: Example simulcast modes with two DRM signals

Figure K.5 illustrates the possible use of two half channel DRM signals.

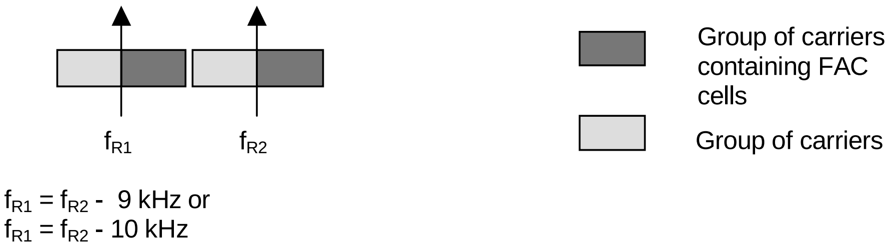


Figure K.6: Example of base/enhancement DRM signals

Figure K.6 illustrates the possible use of two DRM signals as a combination of a base layer and enhancement layer for a multiplex. In this case the AFS data entities signal the frequency of the other layer. All receivers will be able to decode the base layer, whilst some will also be able to decode the enhancement layer and provide higher quality.

Annex L (informative): Pilot reference illustrations

The figures below show the position of the gain reference cells (character "O") for nominal channel bandwidths of up to 10 kHz (spectrum occupancy parameter = 0, 1, 2 or 3). The patterns continue to the right for the 18 kHz and 20 kHz nominal channel bandwidth options. (Spectrum occupancy parameter = 4 or 5).

Robustness mode A:

```
+-> carriers (k)
|
v
symbols (s)                                    DC
                   negative frequencies         :          positive frequencies
    1..1                                        :                                        1..1
    1..0..33333333322222222221111111111000000000:00000000011111111112222222222333333333..0..1
    4..2..87654321098765432109876543210987654321:12345678901234567890123456789012345678..2..4
                                                :
 0  .  .  O...................O................:::O...................O................  O  .
 1  O  .  ....O...................O............:::....O...................O............  .  .
 2  .  .  ........O...................O........:::........O...................O........  .  .
 3  .  .  ............O...................O....:::............O...................O....  .  O
 4  .  O  ................O...................O:::................O...................O  O  .
 5  .  .  O...................O................:::O...................O................  .  .
 6  O  .  ....O...................O............:::....O...................O............  .  .
 7  .  .  ........O...................O........:::........O...................O........  .  .
 8  .  .  ............O...................O....:::............O...................O....  .  O
 9  .  O  ................O...................O:::................O...................O  O  .
10  .  .  O...................O................:::O...................O................  .  .
11  O  .  ....O...................O............:::....O...................O............  .  .
12  .  .  ........O...................O........:::........O...................O........  .  .
13  .  .  ............O...................O....:::............O...................O....  .  O
14  .  O  ................O...................O:::................O...................O  O  .
```

Robustness mode B:

```
+-> carriers (k)
|
v
symbols (s)                                    DC
                   negative frequencies         :          positive frequencies
    1                                           :                                           1
    0..9..33333333322222222221111111111000000000:00000000011111111112222222222333333333..9..0
    3..1..87654321098765432109876543210987654321:12345678901234567890123456789012345678..1..3
                                                :
 0  .  .  ...O.....O.....O.....O.....O.....O....:O.....O.....O.....O.....O.....O.....O.  O  O
 1  .  .  .....O.....O.....O.....O.....O.....O..:..O.....O.....O.....O.....O.....O.....  .  .
 2  O  O  .O.....O.....O.....O.....O.....O.....O:....O.....O.....O.....O.....O.....O...  .  .
 3  .  .  ...O.....O.....O.....O.....O.....O....:O.....O.....O.....O.....O.....O.....O.  O  O
 4  .  .  .....O.....O.....O.....O.....O.....O..:..O.....O.....O.....O.....O.....O.....  .  .
 5  O  O  .O.....O.....O.....O.....O.....O.....O:....O.....O.....O.....O.....O.....O...  .  .
 6  .  .  ...O.....O.....O.....O.....O.....O....:O.....O.....O.....O.....O.....O.....O.  O  O
 7  .  .  .....O.....O.....O.....O.....O.....O..:..O.....O.....O.....O.....O.....O.....  .  .
 8  O  O  .O.....O.....O.....O.....O.....O.....O:....O.....O.....O.....O.....O.....O...  .  .
 9  .  .  ...O.....O.....O.....O.....O.....O....:O.....O.....O.....O.....O.....O.....O.  O  O
10  .  .  .....O.....O.....O.....O.....O.....O..:..O.....O.....O.....O.....O.....O.....  .  .
11  O  O  .O.....O.....O.....O.....O.....O.....O:....O.....O.....O.....O.....O.....O...  .  .
12  .  .  ...O.....O.....O.....O.....O.....O....:O.....O.....O.....O.....O.....O.....O.  O  O
13  .  .  .....O.....O.....O.....O.....O.....O..:..O.....O.....O.....O.....O.....O.....  .  .
14  O  O  .O.....O.....O.....O.....O.....O.....O:....O.....O.....O.....O.....O.....O...  .  .
```

Robustness mode C:

```
+-> carriers (k)
|
v
symbols (s)                                    DC
                   negative frequencies         :          positive frequencies
                                                :
    6.....33333333322222222221111111111000000000:000000000111111111122222222223333333333.....6
    9.....87654321098765432109876543210987654321:12345678901234567890123456789012345678.....9
                                                :
 0  .     ...o...o...o...o...o...o...o...o...o..:o...o...o...o...o...o...o...o...o...o.     o
 1  o     .o...o...o...o...o...o...o...o...o...o:..o...o...o...o...o...o...o...o...o...     .
 2  .     ...o...o...o...o...o...o...o...o...o..:o...o...o...o...o...o...o...o...o...o.     o
 3  o     .o...o...o...o...o...o...o...o...o...o:..o...o...o...o...o...o...o...o...o...     .
 4  .     ...o...o...o...o...o...o...o...o...o..:o...o...o...o...o...o...o...o...o...o.     o
 5  o     .o...o...o...o...o...o...o...o...o...o:..o...o...o...o...o...o...o...o...o...     .
 6  .     ...o...o...o...o...o...o...o...o...o..:o...o...o...o...o...o...o...o...o...o.     o
 7  o     .o...o...o...o...o...o...o...o...o...o:..o...o...o...o...o...o...o...o...o...     .
 8  .     ...o...o...o...o...o...o...o...o...o..:o...o...o...o...o...o...o...o...o...o.     o
 9  o     .o...o...o...o...o...o...o...o...o...o:..o...o...o...o...o...o...o...o...o...     .
10  .     ...o...o...o...o...o...o...o...o...o..:o...o...o...o...o...o...o...o...o...o.     o
11  o     .o...o...o...o...o...o...o...o...o...o:..o...o...o...o...o...o...o...o...o...     .
12  .     ...o...o...o...o...o...o...o...o...o..:o...o...o...o...o...o...o...o...o...o.     o
13  o     .o...o...o...o...o...o...o...o...o...o:..o...o...o...o...o...o...o...o...o...     .
14  .     ...o...o...o...o...o...o...o...o...o..:o...o...o...o...o...o...o...o...o...o.     o
15  o     .o...o...o...o...o...o...o...o...o...o:..o...o...o...o...o...o...o...o...o...     .
16  .     ...o...o...o...o...o...o...o...o...o..:o...o...o...o...o...o...o...o...o...o.     o
17  o     .o...o...o...o...o...o...o...o...o...o:..o...o...o...o...o...o...o...o...o...     .
18  .     ...o...o...o...o...o...o...o...o...o..:o...o...o...o...o...o...o...o...o...o.     o
19  o     .o...o...o...o...o...o...o...o...o...o:..o...o...o...o...o...o...o...o...o...     .
```

Robustness mode D:

```
+-> carriers (k)
|
v
symbols (s)                                    DC
                   negative frequencies         :          positive frequencies
:
    44444333333333322222222221111111111000000000:00000000011111111112222222222333333333344444
    43210987654321098765432109876543210987654321:12345678901234567890123456789012345678901234
                                                :
 0  o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.:o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.
 1  .o..o..o..o..o..o..o..o..o..o..o..o..o..o..o:.o..o..o..o..o..o..o..o..o..o..o..o..o..o..o
 2  ..o..o..o..o..o..o..o..o..o..o..o..o..o..o..:..o..o..o..o..o..o..o..o..o..o..o..o..o..o..
 3  o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.:o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.
 4  .o..o..o..o..o..o..o..o..o..o..o..o..o..o..o:.o..o..o..o..o..o..o..o..o..o..o..o..o..o..o
 5  ..o..o..o..o..o..o..o..o..o..o..o..o..o..o..:..o..o..o..o..o..o..o..o..o..o..o..o..o..o..
 6  o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.:o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.
 7  .o..o..o..o..o..o..o..o..o..o..o..o..o..o..o:.o..o..o..o..o..o..o..o..o..o..o..o..o..o..o
 8  ..o..o..o..o..o..o..o..o..o..o..o..o..o..o..:..o..o..o..o..o..o..o..o..o..o..o..o..o..o..
 9  o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.:o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.
10  .o..o..o..o..o..o..o..o..o..o..o..o..o..o..o:.o..o..o..o..o..o..o..o..o..o..o..o..o..o..o
11  ..o..o..o..o..o..o..o..o..o..o..o..o..o..o..:..o..o..o..o..o..o..o..o..o..o..o..o..o..o..
12  o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.:o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.
13  .o..o..o..o..o..o..o..o..o..o..o..o..o..o..o:.o..o..o..o..o..o..o..o..o..o..o..o..o..o..o
14  ..o..o..o..o..o..o..o..o..o..o..o..o..o..o..:..o..o..o..o..o..o..o..o..o..o..o..o..o..o..
15  o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.:o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.
16  .o..o..o..o..o..o..o..o..o..o..o..o..o..o..o:.o..o..o..o..o..o..o..o..o..o..o..o..o..o..o
17  ..o..o..o..o..o..o..o..o..o..o..o..o..o..o..:..o..o..o..o..o..o..o..o..o..o..o..o..o..o..
18  o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.:o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.
19  .o..o..o..o..o..o..o..o..o..o..o..o..o..o..o:.o..o..o..o..o..o..o..o..o..o..o..o..o..o..o
20  ..o..o..o..o..o..o..o..o..o..o..o..o..o..o..:..o..o..o..o..o..o..o..o..o..o..o..o..o..o..
21  o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.:o..o..o..o..o..o..o..o..o..o..o..o..o..o..o.
22  .o..o..o..o..o..o..o..o..o..o..o..o..o..o..o:.o..o..o..o..o..o..o..o..o..o..o..o..o..o..o
23  ..o..o..o..o..o..o..o..o..o..o..o..o..o..o..:..o..o..o..o..o..o..o..o..o..o..o..o..o..o..
```

Robustness mode E:

The figure below shows the position of the gain reference cells (character "o") for nominal channel bandwidths of 100 kHz (Spectrum occupancy parameter = 0).

```
+-> carriers (k)
|
v
symbols (s)                                    DC
                   negative frequencies         :         positive frequencies
    1...                                        :                                        ...1
    0.....33333333322222222221111111111000000000:00000000011111111112222222222333333333.....0
    6.....87654321098765432109876543210987654321:12345678901234567890123456789012345678.....6
                                                :
 0  .     ........o...............o...............o...............o...............o....     .
 1  o     ............o...............o...............o...............o...............o     .
 2  .     o...............o...............o...............o...............o............     o
 3  .     ....o...............o...............o...............o...............o........     .
 4  .     ........o...............o...............o...............o...............o....     .
 5  o     ............o...............o...............o...............o...............o     .
 6  .     o...............o...............o...............o...............o............     o
 7  .     ....o...............o...............o...............o...............o........     .
 8  .     ........o...............o...............o...............o...............o....     .
 9  o     ............o...............o...............o...............o...............o     .
10  .     o...............o...............o...............o...............o............     o
11  .     ....o...............o...............o...............o...............o........     .
12  .     ........o...............o...............o...............o...............o....     .
13  o     ............o...............o...............o...............o...............o     .
14  .     o...............o...............o...............o...............o............     o
15  .     ....o...............o...............o...............o...............o........     .
.
.
39  .     ....o...............o...............o...............o...............o........     .
```

The figures below show the position of the gain reference cells (character "o"), the frequency reference cells (character "f") and FAC cells (character "x") for the DC to 4,5 kHz (nominal).

Robustness mode A: positions for pilot cells

```
+-> carriers (k)
|
v
symbols (s)

DC (not used)
      :                                                                                                   111
      :000000000111111111122222222223333333333444444444455555555556666666666777777777788888888889999999999000
      :123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012
      :
 0    ::o...............f...o...................o...........f.......o.........f.........o...................o
 1    ::....o...........f.......o...................o.......f...........o.....f.............o................
 2    ::........o.......f.......x...o...............x...o...f...........x...o.f.............x...o............
 3    ::........x...o...f...........x...o...............x...f...............x.f.o...............x...o........
 4    ::............x...f...x...........x...o...............f...o...x.........f.x...o...............x...o....
 5    ::o...............f...o...x...........x...o...........f...x...o...x.....f.....x...o...................o
 6    ::....o...........f...x...o...x...........x...o.......f.......x...o...x.f.........x...o................
 7    ::........o.......f.......x...o...x...........x...o...f...........x...o.f.x...........x...o............
 8    ::........x...o...f...........x...o...x...........x...f...x...........x.f.o...x...........x...o........
 9    ::............x...f...x...........x...o...x...........f...o...x.........f.x...o...x...........x...o....
10    ::o...............f...o...x...........x...o...x.......f.......o...x.....f.........o...x...............o
11    ::....o...x.......f.......o...x...............o...x...f...........o...x.f.............o...x............
12    ::........o...x...f...........o...x...............o...f...............o.f.x...............o...x........
13    ::............o...f...............o...x...............f...x.............f.o...x...............o........
14    ::................f...................o...............f...o.............f.....o...................o....
```

Robustness mode B: positions for pilot cells

```
+-> carriers (k)
|
v
symbols (s)

DC (not used)
     :
     :0000000001111111111222222222233333333334444444444555555555566666666667777777777888888888899
     :1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901
     :
 0   :o.....o.....o..f..o.....o.....o.....o.....o....fo.....o.....o..f..o.....o.....o.....o.....o
 1   :..o.....o.....of....o.....o.....o.....o.....o..f..o.....o.....of....o.....o.....o.....o....
 2   :....o.....o.x..fo.....o.x...o.....o.....o.x...of....o.x...o....fo.x...o.....o.....o.....o..
 3   :o.....o.....o.xf..o.....o.x...o.....o.....o.x..fo.....o.x...o..f..o.x...o.....o.....o.....o
 4   :..o.....o.....ofx...o.....o.x...o.....o.....o.xf..o.....o.x...of....o.x...o.....o.....o....
 5   :....o.....o....fo.x...o.....o.x...o.....o.....ofx...o.....o.x..fo.....o.x...o.....o.....o..
 6   :o.....o.x...o..f..o.x...o.....o.x...o.....o....fo.x...o.....o.xf..o.....o.x...o.....o.....o
 7   :..o.....o.x...of....o.x...o.....o.x...o.....o..f..o.x...o.....ofx...o.....o.x...o.....o....
 8   :....o.....o.x..fo.....o.x...o.....o.x...o.....of....o.x...o....fo.x...o.....o.x...o.....o..
 9   :o.....o.....o.xf..o.....o.x...o.....o.x...o....fo.....o.x...o..f..o.x...o.....o.x...o.....o
10   :..o.....o.....ofx...o.....o.x...o.....o.x...o..f..o.....o.x...of....o.x...o.....o.x...o....
11   :....o.....o....fo.x...o.....o.x...o.....o.x...of....o.....o.x..fo.....o.x...o.....o.....o..
12   :o.....o.....o..f..o.x...o.....o.x...o.....o.x..fo.....o.....o.xf..o.....o.x...o.....o.....o
13   :..o.....o.....of....o.x...o.....o.x...o.....o.xf..o.....o.....ofx...o.....o.x...o.....o....
14   :....o.....o....fo.....o.....o.....o.....o.....of....o.....o....fo.....o.....o.....o.....o..
```

Robustness mode C: positions for pilot cells

```
+-> carriers (k)
|
v
symbols (s)

DC (not used)
     :
     :000000000111111111122222222223333333333444444444455555555556666666666
     :123456789012345678901234567890123456789012345678901234567890123456789
     :
 0   :o...o...o.f.o...o...o...o...o...f...o...o..fo...o...o...o...o...o...o
 1   :..o...o...f...o...o...o...o...o.f.o...o...of..o...o...o...o...o...o..
 2   :o...o...o.f.o...o...o...o...o...f...o...o..fo...o...o...o...o...o...o
 3   :..o...o.x.f...o...o.x.o...o...o.f.o...o...ofx.o...o...o.x.o...o...o..
 4   :o...o...o.f.o...o...o.x.o...o...f.x.o...o..fo.x.o...o...o...o...o...o
 5   :..o...o...f.x.o...o...o.x.o...o.f.o.x.o...of..o.x.o...o...o...o...o..
 6   :o...o...o.f.o.x.o...o...o.x.o...f...o.x.o..fo...o.x.o...o...o...o...o
 7   :..o.x.o...f...o.x.o...o...o.x.o.f.o...o.x.of..o...o.x.o...o...o...o..
 8   :o...o.x.o.f.o...o.x.o...o...o.x.f...o...o.xfo...o...o.x.o...o...o...o
 9   :..o...o.x.f...o...o.x.o...o...o.f.o...o...ofx.o...o...o.x.o...o...o..
10   :o...o...o.f.o...o...o.x.o...o...f.x.o...o..fox..o...o...o...o...o...o
11   :..o...o...f.x.o...o...o.x.o...o.f.o.x.o...of..o.x.o...o...o...o...o..
12   :o...o...o.f.o.x.o...o...o.x.o...f...o.x.o..fo...o.x.o...o...o...o...o
13   :..o.x.o...f...o.x.o...o...o.x.o.f.o...o.x.of..o...o.x.o...o...o...o..
14   :o...o.x.o.f.o...o.x.o...o...o.x.f...o...o.xfo...o...o.x.o...o...o...o
15   :..o...o.x.f...o...o.x.o...o...o.f.o...o...ofx.o...o...o.x.o...o...o..
16   :o...o...o.f.o...o...o.x.o...o...f.x.o...o..fo.x.o...o...o...o...o...o
17   :..o...o...f.x.o...o...o.x.o...o.f.o.x.o...of..o.x.o...o...o...o...o..
18   :o...o...o.f.o.x.o...o...o.x.o...f...o.x.o..fo...o.x.o...o...o...o...o
19   :..o...o...f...o...o...o...o...o.f.o...o...of..o...o...o...o...o...o..
```

Robustness mode D: positions for pilot cells

```
+-> carriers (k)
|
v
symbols (s)

DC (not used)
     :
     :00000000011111111112222222222333333333344444
     :12345678901234567890123456789012345678901234
     :
 0   :o..o..f..o..o..o..o.fo..o..f..o..o..o..o..o.
 1   :.o..o.fo..o..o..o..of.o..o.fo..o..o..o..o..o
 2   :..o..of.o..o..o..o..f..o..of.o..o..o..o..o..
 3   :o..o..f.xo..o..o.xo.fo..o.xf..o..o..o..o..o.
 4   :.o..o.fo.xo..o..o.xof.o..o.fo..o..o..o..o..o
 5   :..o..of.o.xo..o..o.xf..o..ofxo..o..o..o..o..
 6   :o..o..f..o.xo..o..o.fo..o..f.xo..o..o..o..o.
 7   :.o..o.fo..o.xo..o..ofxo..o.fo.xo..o..o..o..o
 8   :..o.xof.o..o.xo..o..f.xo..of.o.xo..o..o..o..
 9   :o..o.xf..o..o.xo..o.fo.xo..f..o.xo..o..o..o.
10   :.o..o.fo..o..o.xo..of.o.xo.fo..o.xo..o..o..o
11   :..o..ofxo..o..o.xo..f..o.xof.o..o.xo..o..o..
12   :o..o..f.xo..o..o.xo.fo..o.xf..o..o.xo..o..o.
13   :.o..o.fo.xo..o..o.xof.o..o.fo..o..o.xo..o..o
14   :..o..of.o.xo..o..o.xf..o..ofxo..o..o..o..o..
15   :o..o..f..o.xo..o..o.fo..o..f.xo..o..o..o..o.
16   :.o..o.fo..o.xo..o..ofxo..o.fo.xo..o..o..o..o
17   :..o.xof.o..o.xo..o..f.xo..of.o.xo..o..o..o..
18   :o..o.xf..o..o.xo..o.fo.xo..f..o.xo..o..o..o.
19   :.o..o.fo..o..o.xo..of.o.xo.fo..o.xo..o..o..o
20   :..o..ofxo..o..o.xo..f..o.xof.o..o.xo..o..o..
21   :o..o..f.xo..o..o.xo.fo..o.xf..o..o.xo..o..o.
22   :.o..o.fo.xo..o..o.xof.o..o.fo..o..o.xo..o..o
23   :..o..of.o..o..o..o..f..o..of.o..o..o..o..o..
```

Robustness mode E:

The figure below shows the position of the gain reference cells (character "o") and the FAC cells (character "x") for the signal without presenting the carriers around DC.

```
+-> carriers (k)
|
v
symbols (s)

                          negative frequencies                      positive frequencies
     1111111                                                                                        1111111
     000000099999999998888888888777777777766666666665      566666666667777777777888888888899999999990000000
     654321098765432109876543210987654321098765432109      901234567890123456789012345678901234567890123456
    :
 0  :............o...............o...............o...      .......o...............o...............o........
 1  :o...............o...............o...............      ...........o...............o...............o....
 2  :....o...............o...............o...........      ...............o...............o...............o
 3  :........o...............o...............o.......      ...o...............o...............o............
 4  :a...a...a...o...a...a...a...o...a...a...a...o...      ...a...o...a...a...a...o...a...a...a...o...a...a
 5  :o...............o...........x...o...........x...      .......x...o...........x...o...............o....
 6  :....o...........x...o...........x...o...........      ...........x...o...........x...o...............o
 7  :........o...........x...o...........x...o.......      ...o...........x...o...........x...o............
 8  :............o...........x...o...........x...o...      ...x...o...........x...o...............o........
 9  :o...............o...........x...o...........x...      .......x...o...........x...o...............o....
10  :....o...........x...o...........x...o...........      ...........x...o...........x...o...............o
11  :........o...........x...o...........x...o.......      ...o...........x...o...........x...o............
12  :............o...........x...o...........x...o...      ...x...o...........x...o...............o........
13  :o...............o...........x...o...........x...      .......x...o...........x...o...............o....
14  :....o...........x...o...........x...o...........      ...........x...o...........x...o...............o
15  :........o...........x...o...........x...o.......      ...o...........x...o...........x...o............
16  :............o...........x...o...........x...o...      ...x...o...........x...o...............o........
17  :o...............o...........x...o...........x...      .......x...o...........x...o...............o....
18  :....o...........x...o...........x...o...........      ...........x...o...........x...o...............o
19  :........o...........x...o...........x...o.......      ...o...........x...o...........x...o............
20  :............o...........x...o...........x...o...      ...x...o...........x...o...............o........
21  :o...............o...........x...o...........x...      .......x...o...........x...o...............o....
22  :....o...........x...o...........x...o...........      ...........x...o...........x...o...............o
23  :........o...........x...o...........x...o.......      ...o...........x...o...........x...o............
24  :............o...........x...o...........x...o...      ...x...o...........x...o...............o........
25  :o...............o...........x...o...........x...      .......x...o...........x...o...............o....
26  :....o...........x...o...........x...o...........      ...............o...............o...............o
27  :........o...............o...............o.......      ...o...............o...............o............
28  :............o...............o...............o...      .......o...............o...............o........
29  :o...............o...............o...............      ...........o...............o...............o....
30  :....o...............o...............o...........      ...............o...............o...............o
31  :........o...............o...............o.......      ...o...............o...............o............
32  :............o...............o...............o...      .......o...............o...............o........
33  :o...............o...............o...............      ...........o...............o...............o....
34  :....o...............o...............o...........      ...............o...............o...............o
35  :........o...............o...............o.......      ...o...............o...............o............
36  :............o...............o...............o...      .......o...............o...............o........
37  :o...............o...............o...............      ...........o...............o...............o....
38  :....o...............o...............o...........      ...............o...............o...............o
39  :a...a...o...a...a...a...o...a...a...a...o...a...      ...o...a...a...a...o...a...a...a...o...a...a...a
```

Robustness mode E:

The figure below shows the position of the the AFS cells (character "a") and SDC cells (character "~") at the transmission super-frame boundary - the last 7 symbols of transmission frame 4 and the first 7 symbols of transmission frame 1 are shown - and includes the gain reference cells (character "o"), the FAC cells (character "x") for the signal without presenting the carriers around DC.

```
+-> carriers (k)
|
v
symbols (s)

                         negative frequencies                                 positive frequencies
     1111111                                                                                                      1111111
     000000099999999998888888888777777777766666666665      566666666667777777777888888888899999999990000000
     654321098765432109876543210987654321098765432109      901234567890123456789012345678901234567890123456
    :
33  :o...............o...............o...............      ...........o...............o...............o....   | f
34  :....o...............o...............o...........      ...............o...............o...............o   | r
35  :........o...............o...............o.......      ...o...............o...............o............   | a
36  :............o...............o...............o...      .......o...............o...............o........   | m
37  :o...............o...............o...............      ...........o...............o...............o....   | e
38  :....o...............o...............o...........      ...............o...............o...............o   | 4
39  :a...a...o...a...a...a...o...a...a...a...o...a...      ...o...a...a...a...o...a...a...a...o...a...a...a  /
 0  :~~~~~~~~~~~~o~~~~~~~~~~~~~~~o~~~~~~~~~~~~~~~o~~~      ~~~~~~~o~~~~~~~~~~~~~~~o~~~~~~~~~~~~~~~o~~~~~~~~  \
 1  :o~~~~~~~~~~~~~~~o~~~~~~~~~~~~~~~o~~~~~~~~~~~~~~~      ~~~~~~~~~~~o~~~~~~~~~~~~~~~o~~~~~~~~~~~~~~~o~~~~   | f
 2  :~~~~o~~~~~~~~~~~~~~~o~~~~~~~~~~~~~~~o~~~~~~~~~~~      ~~~~~~~~~~~~~~~o~~~~~~~~~~~~~~~o~~~~~~~~~~~~~~~o   | r
 3  :~~~~~~~~o~~~~~~~~~~~~~~~o~~~~~~~~~~~~~~~o~~~~~~~      ~~~o~~~~~~~~~~~~~~~o~~~~~~~~~~~~~~~o~~~~~~~~~~~~   | a
 4  :a~~~a~~~a~~~o~~~a~~~a~~~a~~~o~~~a~~~a~~~a~~~o~~~      ~~~a~~~o~~~a~~~a~~~a~~~o~~~a~~~a~~~a~~~o~~~a~~~a   | m
 5  :o...............o...........x...o...........x...      .......x...o...........x...o...............o....   | e
 6  :....o...........x...o...........x...o...........      ...........x...o...........x...o...............o   | 1
```

Annex M (informative): MSC configuration examples

The examples below demonstrate some possibilities for configuring the MSC. Especially the mapping from services to audio or data streams is covered together with some limitations which need to be respected when assembling the DRM multiplex.

General Preface:

- The DRM multiplex may contain up to four streams in the MSC, each carrying audio or data information.
- An audio stream is described by SDC data entity type 9. A synchronous data stream is described by SDC data entity type 5. A packet mode data stream ("data-pm") consists of 1 to 4 "sub-streams" (distinguished by their packet id), each carrying an individual data application described by SDC data entity type 5.
- 1 to 4 services can be signalled to the user. A data service points to one data (sub-)stream. An audio service points to one audio stream optionally carrying text message information ("tm") and/or one to four data (sub-streams.
- Audio services are mapped to audio streams by SDC data entity type 9. Data (and audio) services are mapped to data streams by SDC data entity type 5.
- If several services point to the same (sub-)stream, the stream configurations in SDC data entity type 5 or 9 need to be identical.

NOTE 1: The short id values used for the services below are example values only. The short id values from the range 0-3 may be assigned to the services of a DRM multiplex in any order. A DRM multiplex therefore may carry one single service only using short id 3.

EXAMPLE 1: A very simple DRM multiplex consists of just one single audio service pointing to the one and only audio stream. The audio stream may contain text messages.

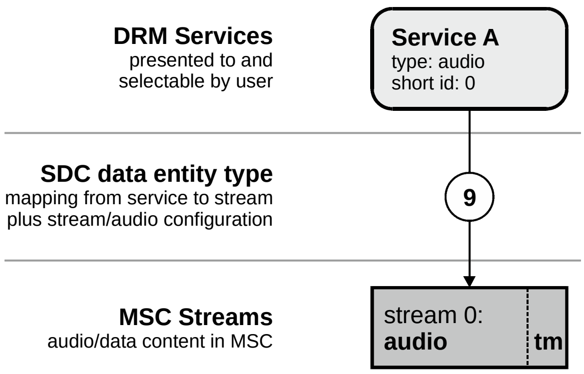


Figure M.1: Example of simple DRM multiplex with an audio service

EXAMPLE 2: An equally simple DRM multiplex consists of just one single data service pointing to the one and only data stream. This DRM multiplex consequently does not provide any audio programme, but exclusively carries a data service.

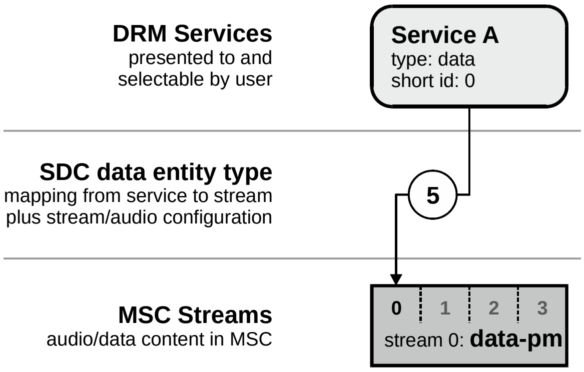


Figure M.2: Example of simple DRM multiplex with a data service

EXAMPLE 3: There are two audio services. Both point to the same audio stream. One of these services ("Service B") points to an additional packet mode data sub-stream carrying a multimedia data application. In total there are one audio stream and one data stream using packet mode.

Note that if multiple services point to the same stream the configuration for that stream (carried in SDC data entities type 9 for audio information or in SDC data entities type 5 for data application information) need to be the same. So in this example both or none of the audio services can carry a text message.

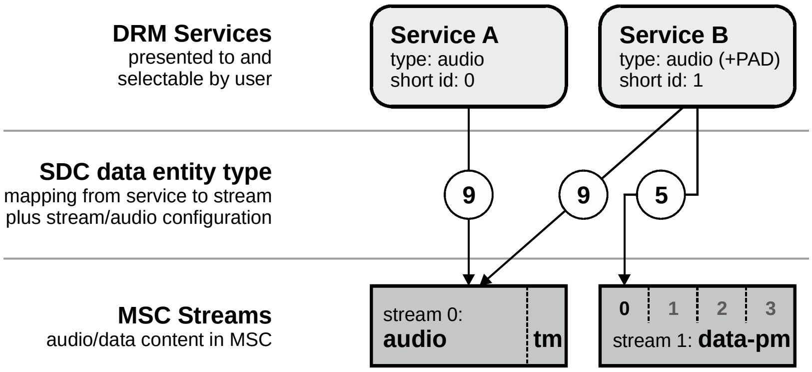


Figure M.3: Example of DRM multiplex with two streams

EXAMPLE 4: There is one audio service and one data service signalled to the user. The DRM multiplex consists of one audio stream and one data stream in packet mode containing one sub-stream. The data service points to the data application carried in the one sub-stream of the data stream. The audio service points to the audio stream and additionally also to the one sub-stream of the data stream.

Note that in this case the data application description in the two SDC data entities type 5 need to be identical for both services referencing the same packet mode sub-stream.

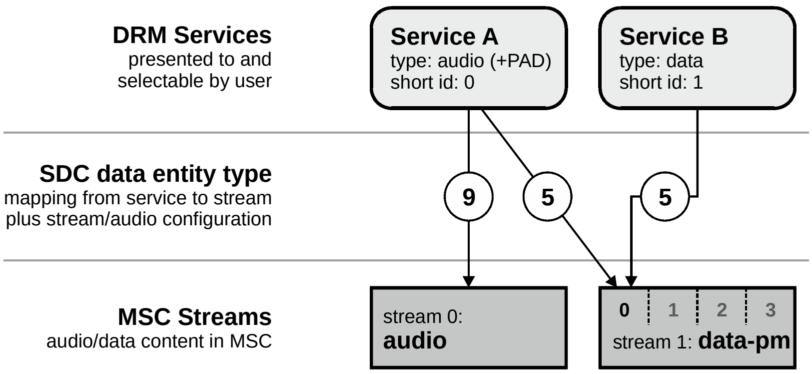


Figure M.4: Example of DRM multiplex with two streams

EXAMPLE 5: There is one audio service and one data service signalled to the user. The DRM multiplex consists of one audio stream and two data streams in packet mode containing a total of four sub-streams. The data service points to the data application carried in sub-stream 0 of the data stream 0. The audio service points to the audio stream and additionally also to all four sub-streams of the 2 data streams, thereby providing 4 data applications along with the audio service.

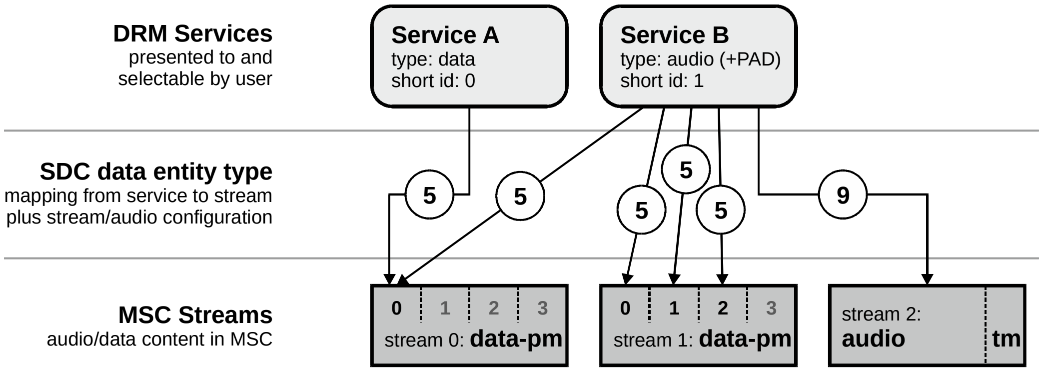


Figure M.5: Example of DRM multiplex with three streams

EXAMPLE 6: To make close to maximum use of the possibilities of a DRM multiplex, the following scenario is possible:

There are three audio services each pointing to its own audio stream. Each of these three audio streams carries text messages. Two of these audio services point to their own data application carried as a sub-stream of a packet mode data stream (being the fourth stream in the DRM multiplex), while the third audio service points to two data applications carried in the remaining sub-streams. There is also a data service pointing to the fourth sub-stream of the data stream.

In total there are:

- three audio services;
- one data service.

These four services point to 10 different "logical channels":

- three different audio streams with their own text messages;
- one data stream in packet mode with four sub-streams.

NOTE 2: The packet mode configuration parameters (e.g. the packet length) of all five SDC data entities type 5 (describing the four data applications and the data service carried in the packet mode data stream) are identical!

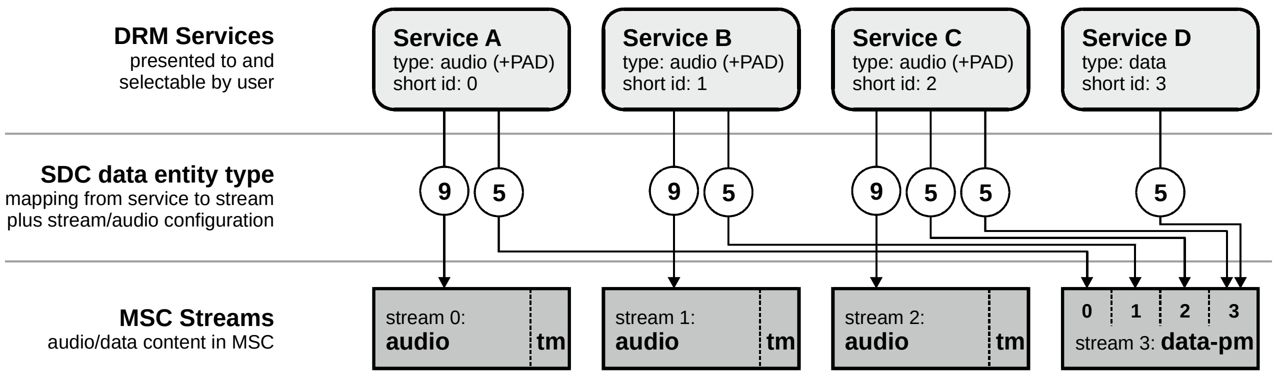


Figure M.6: Example of DRM multiplex with four streams

Annex N (informative): Signalling Warning/Alarm announcements

DRM permits various types of announcements to be made. The signalling information is provided by the SDC data entity type 6, see clause 6.4.3.7 and clause F.5.

For Warning/Alarm announcements only, additional signalling is provided in the FAC to allow a receiver to determine whether an active Warning/Alarm announcement is in progress without having to decode the SDC.

This permits a lower power implementation for receivers that periodically wake up from standby to assess conditions for full switch-on to deliver emergency broadcasts.

During the duration of an active Warning/Alarm announcement carried on any of the services in the DRM multiplex, the **service descriptor** field in the FAC service parameters of all services in the multiplex is set to the special value of 30.

NOTE: Usually the **service descriptor** field describes an individual service to enable receivers to scan for features of interest; however, the value 30 represents the status of the overall multiplex (i.e. that at least one service is carrying an active Warning/Alarm announcement).

Therefore, for the duration of an active Warning/Alarm announcement, the **programme type** of audio services and the **application identifier** of data services will not be available.

Receivers that stored the **programme type** value for an audio service before the Warning/Alarm announcement was activated may continue using this service-specific value for the duration of the active Warning/Alarm announcement.

For data services, SDC data entity type 5 provides all the required service signalling parameters for identification and decoding.

Annex O (normative): Interpretation of schedules for Alternative Frequency Signalling

The "Alternative frequency signalling: Schedule definition data entity - type 4" provides the functionality to restrict the availability of a list of alternative frequencies to certain time intervals based on a weekly schedule.

In every SDC data entity type 4 the following information can be signalled:

- With the Day Code field it can be indicated to which days of the week (Monday to Sunday) the following time range shall apply. Any day-combination from 1 to 7 days can be signalled.
- Using the Start Time and the Duration value, a time interval can be specified. This time interval applies to all specified days-of-the-week (using the Day Code).
  The Start Time value indicates the minutes since midnight UTC (for every indicated day of the week), ranging from 00:00 to 23:59.
  The Duration value specifies the number of minutes after (and including) the start time. It can potentially span more than one week. So for example it is possible to cover a full weekend using one single SDC data entity type 4.
- More than one time interval per day or different day-time-combinations can be specified by broadcasting multiple SDC data entities type 4 with the same Schedule Id (using the list mechanism for the version flag).

Every receiver has to evaluate these values in a consistent way. Therefore the following text defines how the receiver has to interpret the SDC "Alternative frequency signalling: Schedule description data entities - type 4". The function (presented in pseudo program code notation) checks whether the current time/date is within a scheduled time interval:

```
// input:  time_in_week (minutes since last Monday 00:00 in UTC;
//            the value is in the range 0 ≤ time_in_week < 60 × 24 × 7);
//         schedule_id (id of the schedule to be checked;
//            the value is in the range 0 ≤ schedule_id <= 15 )
// output: boolean value (true: time_in_week is inside schedule)

bool IsInsideSchedule (time_in_week, schedule_id)
{
   // the schedule_id value 0 is fixed to 'always valid':
   if (schedule_id == 0)
   {
       return true
   }

   for every SDC entity with the given schedule_id
   {
      extract (day_code, start_time, duration) from SDC entity

      for every day specified by day_code
      {
         // minutes_since_monday(day) returns the number of minutes
         // of the start (00:00) of the indicated day since Monday 00:00
         // (it is a multiple of 24 × 60)
         // e.g. Monday    -> 0 × 24 × 60 = 0
         //      Tuesday   -> 1 × 24 × 60 = 1 440
         //      Wednesday -> 2 × 24 × 60 = 2 880, etc.

         schedule_start = minutes_since_monday (day) + start_time
         schedule_end   = schedule_start + duration

         // the normal check (are we inside start and end?):
         if (time_in_week >= schedule_start AND
             time_in_week <= schedule_end)
         {
             return true
         }

         // the wrap-around check:
         minutes_per_week = 7 × 24 × 60
         if (schedule_end > minutes_per_week)
         {
```

```
            // our duration wraps into next Monday (or even later)
            if (time_in_week < (schedule_end - minutes_per_week))
            {
                return true
            }
        }
    }
}

return false
}
```

The encoding for a certain time interval is not unique. A 48-hour interval starting on Wednesday 10:00 could be encoded as:

- "Wednesday and Thursday; start time 10:00; duration 24 hours".
- "Wednesday; start time 10:00; duration 48 hours".
- or using two SDC data entities with the same schedule id:
  "Wednesday; start time 10:00; duration 24 hours" and "Thursday; start time 10:00; duration 24 hours".
- or using two SDC data entities with the same schedule id:
  "Wednesday; start time 10:00; duration 10 hours" and "Wednesday; start time 20:00; duration 38 hours".

It is up to the encoding side to describe a certain schedule with as few SDC data entities as possible.

Annex P (informative): Transmit diversity

The DRM system is designed for various transmission environments with different delay spread and Doppler spread. Multipath environments with short and strong echoes, which typically occur in urban canyons, lead to a huge coherence bandwidth so that the channel can be described as flat instead of frequency-selective. Systems with a bandwidth smaller than the coherence bandwidth can accordingly suffer from flat fading. This is especially the case for robustness mode E. Time interleaving applied to the DRM system improves the performance of moving receivers in such circumstances.

A further method to overcome flat fading is antenna diversity, which means the application of more than one antenna at the receiver or transmitter. Antenna diversity at the receiver is effective but difficult to implement in small receiver boxes. For broadcast systems the use of transmit diversity is a good alternative or addition to receive diversity.

In the development of robustness mode E, different methods, such as space time coding and delay diversity, were evaluated. This evaluation showed that delay diversity is the preferred choice because space time coding requires more overhead in the signal for channel estimation and it is more sensitive against the time-incoherence for fast fading channels.

The idea of delay diversity is quite simple. In addition to the original signal a delayed version of the same signal is transmitted from another spatially separated antenna. This method increases the channel delay spread by an additional echo with a comparable effect as with single frequency networks. The application of transmit delay diversity does not require any modifications at the receiver.

Figure P.1 shows how delay diversity can be implemented at the transmitter for an arbitrary number of antennas. After the OFDM modulation with an IFFT the signal path is split according to the number of antennas. Each signal path will be delayed by a chosen value δ_k before insertion of the guard interval.

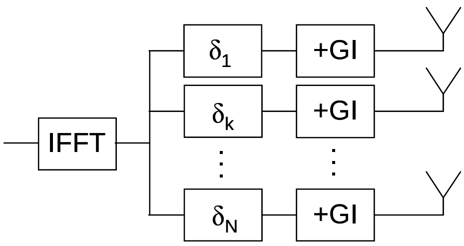


Figure P.1: Transmit delay diversity scheme

The only parameters which have to be chosen are the delay δ_k of each path. Two requirements have to be considered:

- the delay δ_k should be large enough to increase the frequency selectivity of the composed channel that is the superposition of the channels for the transmit antennas;
- and it should be much less than the guard interval duration T_g in order to avoid inter-symbol-interference.

According to the first requirement the value δ should be at least 10 µs for a two antenna system in robustness mode E. This value also fulfils the second requirement because only 4 % of the guard interval duration gets lost. For further optimization the appropriate scientific literature is recommended.

The improvements which can be obtained with transmit delay diversity depend on the actual transmission channel. Simulations have been performed for the channel profiles described in clause B.2. They show that the SNR gain at a BER of 10^{-4} for the Terrain Obstructed profile at 60 km/h receiver speed is around 1 dB, for the Typical Urban profile at 60 km/h around 2 dB and the Typical Urban profile at 5 km/h more than 4 dB.

Annex Q (informative): Seamless reconfiguration

Clause 6.4.6 explains the mechanism used for reconfiguration, which can occur on a transmission super-frame boundary. Depending on the nature of the reconfiguration, the receiver may be able to follow the selected service without audio interruption. Table Q.1 indicates for which type of reconfiguration this is possible, and for which types resynchronization is required.

Table Q.1: Cases of reconfiguration

Case	Reconfiguration type	Possible without audio interruption	Notes
1	Changing the robustness mode	No	Requires full resynchronization
2	Changing the spectrum occupancy	No	Requires full resynchronization
3	Changing the interleaver mode	No	Affects the latency of the service
4	Changing the audio configuration	No	Requires audio decoder resynchronization
5	Changing the data rate of the current audio service	No	Requires audio decoder resynchronization
6	Changing the SDC QAM mode	Yes	Does not affect the MSC
7	Reordering of service multiplex	Yes	If the bitrate and audio configuration of the currently selected service is unaffected
8	Changing the MSC QAM mode	Yes	If the bitrate and audio configuration of the currently selected service is unaffected
9	Changing the MSC protection level	Yes	If the bitrate and audio configuration of the currently selected service is unaffected
NOTE: A reconfiguration that combines more than one reconfiguration type is possible without audio interruption only if all the constituent reconfiguration types are possible without audio interruption.			

In order for seamless reconfiguration to work, several parts of the chain need to behave correctly:

- The modulator or MDI generator should generate the appropriate FAC and SDC signalling to indicate the timing of the reconfiguration and the new parameters (see clause 6.4.6).
- The modulator should generate a continuous DRM signal through the reconfiguration, and should not clear the contents of any buffers or memories unnecessarily because of the change of parameters.
- The receiver should:
  - Interpret correctly the signalling of the reconfiguration.
  - Not clear the contents of any buffers or memories unnecessarily because of the change of parameters.
  - Allow correctly for the delay through the de-interleaver when applying the new parameters.
  - Conceal adverse effects through use of audio muting.

One case where particular care should be taken concerns a change of MSC mode, i.e. the MSC constellation. Figure Q.1 shows the contents of the cell interleaver and de-interleaver following a change from 16-QAM to 64-QAM. For clarity, only the convolutional part of the interleaver, and not the pseudo-random part, is shown. The following points should be noted:

- Both the interleaver and de-interleaver contain a mixture of both types of constellation. The representation used in the interleaver and de-interleaver memories should therefore be able to deal with this mixture.
- In the signal on the air, the MSC cells in a given frame will contain a mixture of both types of constellation.
- For a given multiplex frame, the cells at the output of the de-interleaver are nevertheless all of the same constellation.

- The constellation type at the de-interleaver output will not change until the new constellation cells have worked their way through the de-interleaver, and so the change of parameters for the downstream processing should be delayed accordingly.
- The number of cells in a multiplex frame has not changed, so the interleaver structure remains unchanged.

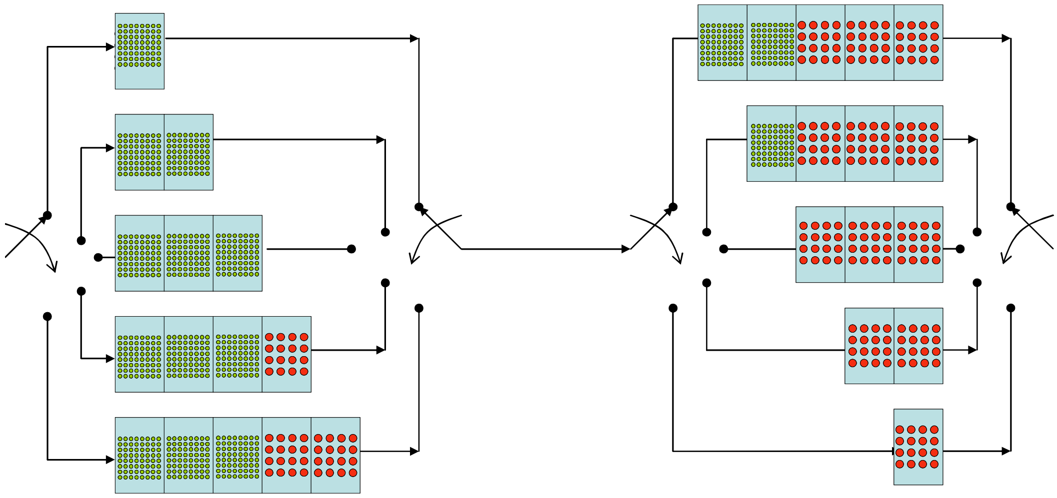


Figure Q.1: Contents of interleaver/de-interleaver system during change of MSC mode

Annex R (informative): Seamless receiver switching between DRM, DAB, AM and FM broadcasts

R.1 Overview

The AFS feature enables a receiver to detect alternative broadcasts of the same audio service (i.e. the same audio content) that is currently presented to the listener. These alternative broadcasts may be available as part of differently configured DRM multiplexes or on other broadcast systems like DAB, AM or FM.

To allow a multi-standard radio receiver to perform seamless switching between all alternative sources of a particular audio programme, both broadcasters and receiver manufacturers need to follow the rules defined in this annex when configuring their transmission networks and designing their receiver devices, respectively.

A typical use case for the seamless receiver switching feature is a broadcaster targeting a mobile audience with a multi-standard broadcast network comprising for example DRM robustness mode E, DAB and FM transmissions. On the receiver side, this feature could be particularly attractive for in-car receivers, which may experience a constantly varying coverage through different broadcast systems.

Both broadcasters and receiver manufacturers are free to support the optional seamless receiver switching feature. However, if enabled or used by either party then the timing requirements laid out in this annex need to be followed to serve as the reliable common reference point for both broadcasters and receiver implementers.

R.2 General network timing considerations

On the receiver side, every broadcast system requires a well-defined minimum processing and decoding time. This decoding delay is mainly defined by mandatory demodulation and decoding steps specific to each digital broadcast system, like de-interleaving, audio super frame management, etc. In contrast, there is basically no system inherent minimum processing delay for analogue broadcast systems like AM and FM. Processing steps in the receiver that are independent from the tuned broadcast system or those that are receiver model specific (like audio post-processing) will not be taken into account for this feature; the receiver is responsible for internally re-aligning those extra delays for all supported broadcast systems.

If all signals-on-air were transmitted simultaneously for all broadcast systems (with respect to the contained audio content), receivers would need to internally buffer and delay the uncompressed audio content received via AM and FM, to be able to seamlessly switch to and from the same audio content available after the time consuming decoding process of digital broadcast systems. This would lead to expensive buffer memory requirements for receivers and therefore needs to be avoided - if any signal buffering is required in the receiver, it should be in the decoding path of the digital broadcast systems, thereby limiting the buffer requirements (if any) to highly compressed audio content.

Therefore the signal-on-air will be delayed on the transmitter side by the broadcaster individually for each broadcast system, so that this signal-on-air has a well-defined delay transmission time difference between each broadcast system (with regard to the contained audio content). This transmission offset between broadcast systems on the broadcast side will be dimensioned such that all mandatory processing steps in the receiver are covered, and that in case any buffering will be performed in the receiver, it lies within the decoding path of the digital broadcast systems instead of the analogue broadcast systems. The benefit of this approach is that receivers are only required to buffer the compressed digital signal (e.g. the relevant compressed audio stream), if any buffering is required at all.

Figure R.1 visualizes this concept.

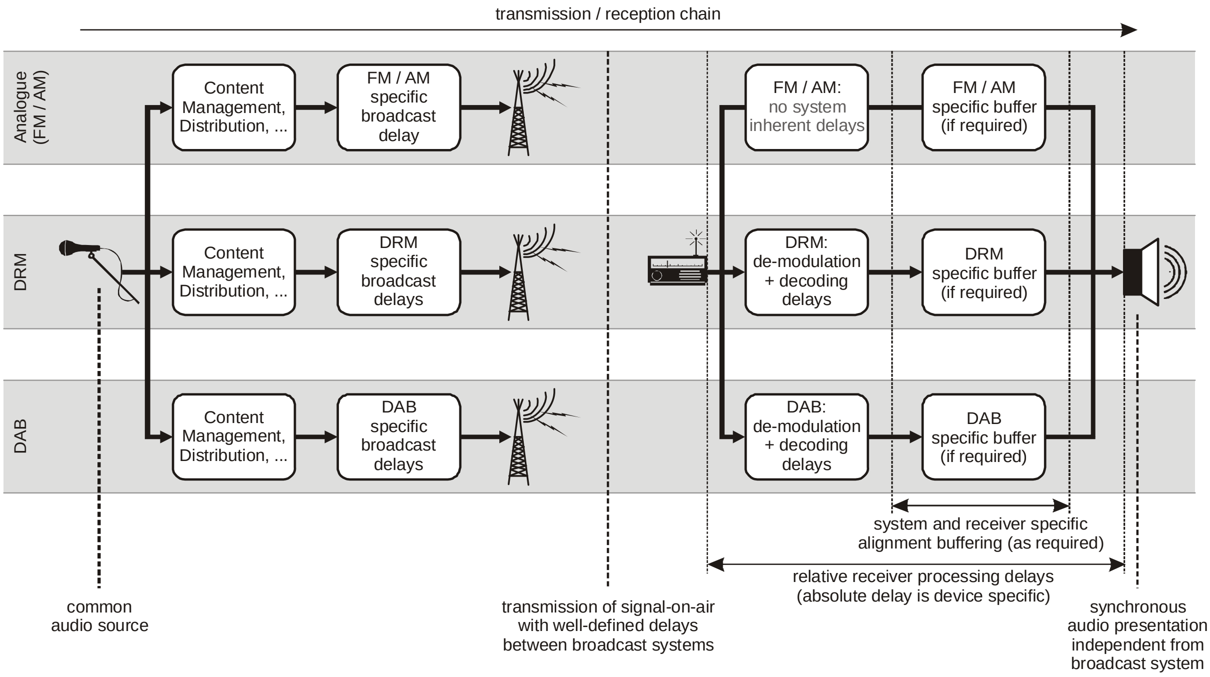


Figure R.1: Broadcast system specific transmission paths to enable seamless receiver switching

NOTE: The seamless receiver switching feature specifies well-defined content delays for the different broadcast systems with regard to the signal-on-air. It is up to the broadcaster to fulfil these timing conditions in cooperation with the involved broadcast equipment manufacturers and network operators. On the receiver side, these timing conditions are to be interpreted and implemented as well-defined relative on-air delay values between the different broadcast systems to result in a co-timed presentation of the same audio content independent from the broadcast system in use. However, it is not the intention of the present document to define or limit the absolute decoding delay introduced by a particular receiver model, as long as any additional decoding delay required by a particular receiver model is applied with equal duration to the decoding paths of all supported broadcast systems.

R.3 Network synchronization rules

This clause defines the exact delay values that need to be introduced to the audio content with regard to the signal-on-air on each broadcast system. The absolute delay values depend on the types of broadcast systems that are part of the broadcaster's network - even if some of those broadcast systems (like DRM robustness modes A, B, C and D with long interleaving) are only included temporarily each day.

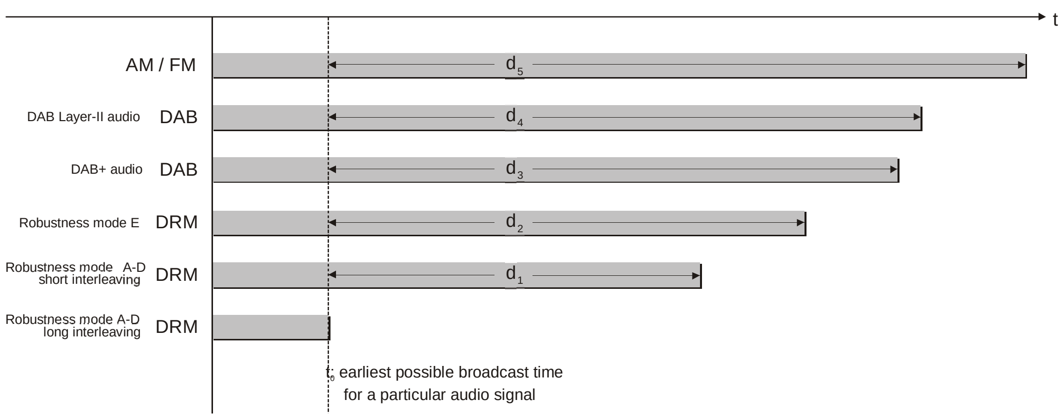


Figure R.2: Transmission delays per broadcast system relative to a common time reference t_0

Figure R.2 describes the delays that need to be introduced for each broadcast system in relation to a common time reference t_0, i.e. the earliest possible broadcast of a particular audio content in the signal-on-air. For example, the DRM transmission frame that carries a particular piece of audio will be broadcast as soon as possible via DRM robustness modes A, B, C and D with long interleaving, while the same audio content over FM/AM will be transmitted over the air with a delay of d_6 relative to the described DRM signal.

The exact delay values d_1 to d_6 will be as defined in table R.1.

Table R.1: Transmission delays for signals-on-air (all broadcast systems utilized)

Broadcast System	Delay of signal-on-air for same audio content relative to common reference t_0
DRM robustness mode A to D, long interleaving	0 ms (common time reference t_0)
DRM robustness mode A to D, short interleaving	d_1 = 1 600 ms
DRM robustness mode E	d_2 = 2 050 ms
DAB (HE AAC v2 audio, "DAB+")	d_3 = 2 450 ms
DAB (DAB Layer-II audio)	d_4 = 2 550 ms
AM/FM	d_5 = 3 000 ms

NOTE: These transmission delay values for audio content of the signal-on-air were derived from the following assumed minimum demodulation/decoding delays in a receiver:
FM/AM: 0 ms; DAB with Layer-II audio: 408 ms; DAB with HE AAC v2 audio (DAB+): 504 ms; DRM robustness mode E: 912 ms; DRM robustness modes A, B, C and D with short interleaver: 1 360 ms; DRM robustness modes A, B, C and D with long interleaver: 2 960 ms.
The derived transmission delay values for the digital broadcast systems were slightly rounded down to allow for some extra processing in the digital reception path without the need to buffer the analogue AM/FM signal in the receiver.

These transmission delay values are valid if *all* defined broadcast systems are used as part of a broadcaster's transmission network. However, if a broadcaster uses only some of these broadcast systems for this service, the absolute overall content delay can be reduced. In this case, the broadcast system with the smallest value, d_x, sets the common reference for the delays of all other broadcast systems being part of the transmission network.

EXAMPLE 1: If a transmission network only comprises DRM robustness mode E and FM transmissions, the DRM transmission can be broadcast without any additional delay (defining the new common reference t_0), while the required delay for the analogue FM broadcast is d_6-d_2 = 950 ms.

EXAMPLE 2: If a transmission network only comprises DRM robustness modes A, B, C and D (short interleaver), DAB+, and AM transmissions, the DRM transmission can be broadcast without any additional delay (defining the new common reference t_0). The required broadcast delay for the DAB+ transmission in this case is d_4-d_1 = 850 ms, the delay for the analogue AM signal is d_6-d_1 = 1 400 ms.

R.4 Receiver implementation rules

It is the responsibility of each receiver implementation to internally align and co-time the decoded audio signal for each reception path and for each signal configuration, based on the well-defined transmission delays of the audio content carried in the signal-on-air as well as any receiver specific decoding and processing steps.

There are additional measures available for receiver implementations to support the listener experience of seamless switching of audio signals from different broadcast systems:

- If a receiver needs to switch between two alternative sources for the same audio programme, it should perform a time limited cross-fading between the two audio signals. This prevents annoying interruptions of the audio signal and covers minimal timing differences between those two sources.

- Alternatively a receiver might calculate the correlation of the decoded audio signals and adjust its internal delay values accordingly. In this case even a direct switch between audio sources should feel seamless.
- In addition the receiver could adjust the audio loudness level of the different sources if possible to support the impression of seamless source switching. If loudness metadata is transmitted alongside the audio programme, the preferred way to ensure the same loudness is to evaluate the embedded loudness information and adjust the loudness appropriately.

R.5 Definition of broadcast signal time references

The following definitions are used as broadcast system specific time references to align the broadcast delays of the signals-on-air. These time references are shown in figure R.2. All of the following definitions refer to the transmission of a particular part of the programme's audio content labelled "A".

NOTE: This definition of broadcast time references does not intend to result in a highly precise broadcast signal alignment as it would for example be required for SFN operation (single frequency networks). Instead these definitions should be seen as reference points that should be targeted by the broadcaster as precisely as technically possible, e.g. by fine-tuning the input delays of audio sources before feeding them into the respective broadcast encoder.

For **analogue AM or FM broadcasts**, the broadcast time shown in figure R.2 refers to the very moment when the audio signal "A" is put on air as part of the AM or FM coded transmission signal, respectively.

For **DRM broadcasts** the indicated broadcast time refers to the start of the transmission frame containing the audio super frame (in case of robustness mode E, the first part of the audio super frame) that carries the audio signal "A" as the first audio samples encoded into that audio super frame.

For **DAB broadcasts using MPEG Audio Layer-II encoding**, the indicated broadcast time refers to the start of the transmission frame starting with the MPEG-II transport stream packet that carries the audio signal "A" as the first audio samples encoded into the audio access unit carried in that transport stream packet.

For **DAB broadcasts using HE AAC v2 encoding ("DAB+")**, the indicated broadcast time refers to the start of the transmission frame starting with the audio access unit that carries the audio signal "A" as the first audio samples encoded into that access unit.

For the digital broadcast systems DAB and DRM, the audio super frame boundaries carrying the audio content "A" should be aligned as closely as possible to meet the timing restrictions described above with respect to the individual audio coding schemes and transmission signal structures.

Annex S (informative): Combined transmission of DRM and FM

A close placement of a robustness mode E signal to an FM signal is possible and can be flexibly configured depending on the existing use of spectrum. In this way, DRM may be introduced into the FM frequency bands.

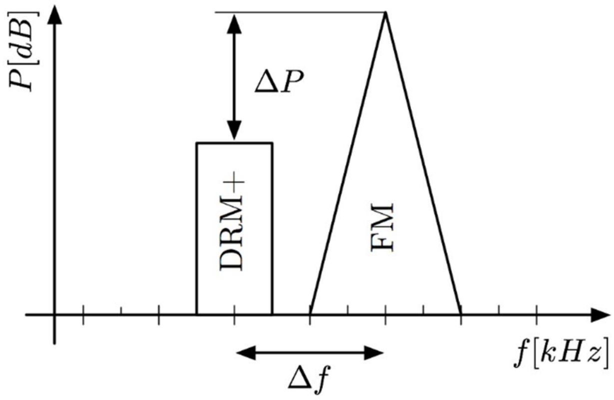


Figure S.1: Example configuration for DRM robustness mode E and FM signal

Figure S.1 shows that the DRM signal can be placed closely to the left or right of the existing FM signal. To guarantee the respective protection levels and audio quality of the FM signal, the carrier frequency distance (Δf) and the power level difference (ΔP) of the FM and the DRM signals can be planned accordingly.

Two transmission configurations are possible: the analogue and digital signals can be combined and transmitted via the same antenna; or the two signals can be transmitted from different antennas.

Different configurations for the DRM signal are possible. The DRM signal can have the same programme as the FM service, a different programme or the same programme and additional programmes. If the same programme is available via DRM and FM, AFS signalling should be sent in the SDC.

Figure S.2 shows some example configurations.

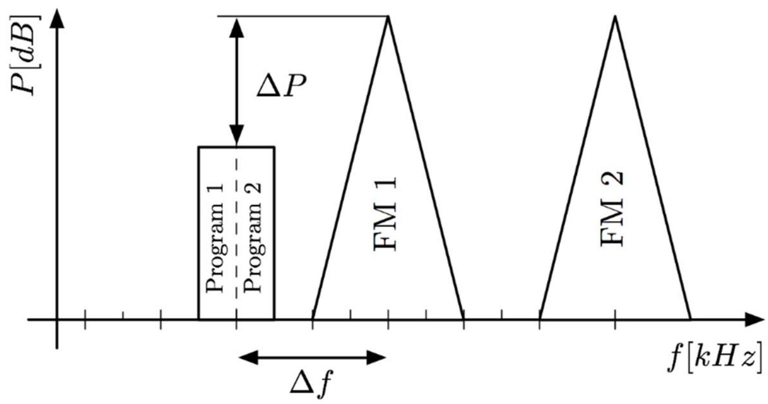


Figure S.2: Example configuration with 2 FM Stations and DRM robustness mode E

Annex T (informative): Assignment of Service identifiers

T.0 Introduction

The Service identifier is defined to be a 24-bit number (see clause 6.3.4). To achieve reliable receiver behaviour, care needs to be taken to ensure that Service identifiers are unique worldwide. This annex describes how to manage the assignment of Service identifiers.

T.1 Domestic services

These services, primarily designed to be consumed in the country of transmission, are generally regulated by national authorities. Therefore, it makes sense for national bodies to allocate Service identifiers to ensure uniqueness. Conflict between Service identifiers would only come from the reception of transmissions from neighbouring countries, or from international services. The former can be dealt with by coordination between countries and the latter can be achieved by ensuring that international services and domestic services are restricted to using particular ranges of the 24-bit coding space.

The RDS system (see IEC 62106 [12]) has allocated Country Codes (CC) and Extended Country Codes (ECC) for most countries. It is recommended that domestic services construct the 24-bit Service identifier by allocating the 8-most significant bits to the ECC, the next 4-bits to the CC(s), and the remaining 12-bits for individual services. In general, this provides a maximum of 4 096 services per country, although some countries have more than one country code (e.g. the USA has 14 providing 57 344 service codes).

To allow for future provision, all codes from 0xA00000 to 0xFFFFFF are designated for allocation using this method (total of 6 291 456 services).

T.2 International services

The High Frequency Coordination Committee (HFCC) and some other bodies coordinate frequencies used in the HF bands for international services. The HFCC has indicated that it will allocate and coordinate DRM Service identifiers for the services for which it provides frequency coordination. The 24-bit code, in the range 0x000000 to 0x9FFFFF (total of 9 437 184 services), should be allocated by a random mechanism. This is because DRM provides service information via the FAC and SDC describing each broadcast in terms of language, label, etc., and it is therefore not necessary to include this information in the Service identifier in any form; indeed doing so causes limitations to the total number of codes that can be allocated because certain combinations will never be used. It is further recommended that no system is applied to the allocation relating the code to any physical characteristics of the broadcast (e.g. DRM mode, frequency, etc.) for the same reasons.

It is recommended that the HFCC (or other issuing body) should generate Service identifier codes randomly from the available stock and allocate them to services. When a service ceases to be transmitted, the code is released back to the issuing body for re-use after a suitable period. The issuing body maintains a central database of codes that have been allocated, but it is not necessary to publish that information.

Broadcasters have the responsibility to check to the best of their ability that no other current broadcast is using the same Service identifier value. In case a clash of identifiers is detected, the involved broadcasters should resolve the situation as quickly as possible to avoid unexpected receiver behaviour.

History

Document history		
V1.1.1	September 2001	Publication as ETSI TS 101 980
V1.2.2	April 2003	Publication
V2.1.1	June 2004	Publication
V2.2.1	October 2005	Publication
V2.3.1	February 2008	Publication
V3.1.1	August 2009	Publication
V3.2.1	June 2012	Publication
V4.1.1	January 2014	Publication
V4.1.2	April 2017	Publication
V4.2.1	January 2021	Publication
V4.3.0	September 2023	Membership Approval Procedure MV 20231103: 2023-09-04 to 2023-11-03