ETSI TS 102 721-3 V1.1.1 (2011-12)

Satellite Earth Stations and Systems;
Air Interface for S-band Mobile Interactive Multimedia (S-MIM);
Part 3: Physical Layer Specification, Return Link
Asynchronous Access

Reference
DTS/SES-00318-3

Keywords
MSS, satellite

ETSI

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

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

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

Contents

Intellectual Property Rights

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

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

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Satellite Earth Stations and Systems (SES).

The present document is part 3 of a multi-part deliverable. Full details of the entire series can be found in part 1 [4].

Introduction

The present document concerns the S-MIM (S-band Mobile Interactive Multimedia) system in which a standardised S-band satellite mobile broadcast system is complemented by the addition of a return channel.

The technology applied has been developed in the framework of the publicly co-funded project "DENISE" (ESTEC / Contract Number 22439/09/NL/US).

The S-MIM system specified herein is designed to provide:

- Interactive mobile broadcast services enhancing DVB-SH services.
- Messaging services for handhelds and vehicular terminals, capable of serving millions of terminals due to a novel optimised radio-interface in the RTN link.
- Real-time emergency services such as voice and file transfer, mainly addressing institutional users on-the-move such as fire brigades, civil protection, etc.

Inside the S-band, the 2 GHz MSS band is of particular interest for interactive multimedia, since it allows two-way transmission. Typically, the DVB-SH standard [i.7] is applied for broadcast transmission; ESDR [i.3] is an alternative. Essential requirements under the R&TTE directive are covered by the harmonized standard EN 302 574 [i.4], [i.5] and [i.6].

1 Scope

The present document is part 3 of the standard and concerns aspects of the air interface for the S-band Mobile Interactive Multimedia (S-MIM) system, and in particular it specifies the Physical Layer for Return Link Asynchronous Access.

The other parts are listed in the foreword of part 1 [4].

2 References

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

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

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

2.1 Normative references

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

[1] ETSI TS 125 212: "Universal Mobile Telecommunications System (UMTS); Multiplexing and channel coding (FDD) (3GPP TS 25.212)".

[2] ETSI TS 125 213: "Universal Mobile Telecommunications System (UMTS); Spreading and modulation (FDD) (3GPP TS 25.213)".

[3] ETSI TS 102 721-6: Satellite Earth Stations and Systems; Radio interface for S-band Mobile Interactive Multimedia (S-MIM); Part 8: "Satellite Earth Stations and Systems; Air Interface for S-band Mobile Interactive Multimedia (S-MIM); Part 6: Protocol Specifications, System Signalling".

[4] ETSI TS 102 721-1: "Satellite Earth Stations and Systems; Air Interface for S-band Mobile Interactive Multimedia (S-MIM); Part 1: General System Architecture and Configurations".

2.2 Informative references

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

[i.1] IEEE Journal on Selected Areas in Communications: "Bandlimited Quasi-Synchronous CDMA: A Novel Satellite Access Technique for Mobile and Personal Communications Systems", R. De Gaudenzi, C. Elia, R. Viola, 1992.

[i.2] Patent: US201 010054131 A 1: "Methods, Apparatuses and System for Asynchronous Spread-Spectrum Communication".

[i.3] ETSI EN 302 550: "Satellite Earth Stations and Systems (SES); Satellite Digital Radio (SDR) Systems; Part 1: Physical Layer of the Radio Interface; Sub-part 1: Outer Physical Layer".

[i.4] ETSI EN 302 574-1: "Satellite Earth Stations and Systems (SES); Harmonized standard for satellite earth stations for MSS operating in the 1 980 MHz to 2 010 MHz (earth-to-space) and 2 170 MHz to 2 200 MHz (space-to-earth) frequency bands; Part 1: Complementary Ground Component (CGC) for wideband systems: Harmonized EN covering the essential requirements of article 3.2 of the R&TTE Directive".

[i.5] ETSI EN 302 574-2: "Satellite Earth Stations and Systems (SES); Harmonized standard for satellite earth stations for MSS operating in the 1 980 MHz to 2 010 MHz (earth-to-space) and 2 170 MHz to 2 200 MHz (space-to-earth) frequency bands; Part 2: User Equipment (UE) for wideband systems: Harmonized EN covering the essential requirements of article 3.2 of the R&TTE Directive".

[i.6] ETSI EN 302 574-3: "Satellite Earth Stations and Systems (SES); Harmonized standard for satellite earth stations for MSS operating in the 1 980 MHz to 2 010 MHz (earth-to-space) and 2 170 MHz to 2 200 MHz (space-to-earth) frequency bands; Part 3: User Equipment (UE) for narrowband systems: Harmonized EN covering the essential requirements of article 3.2 of the R&TTE Directive".

[i.7] ETSI TS 102 585: "Digital Video Broadcasting (DVB); System Specifications for Satellite services to Handheld devices (SH) below 3 GHz".

3 Definitions and abbreviations

3.1 Definitions

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

2 GHz MSS band: 1 980 to 2 010 MHz (earth-to-space) and 2 170 to 2 200 MHz (space-to-earth) frequency bands

NOTE: These paired bands are assigned to MSS.

architecture: abstract representation of a communications system

NOTE: Three complementary types of architecture are defined:

- Functional Architecture: the discrete functional elements of the system and the associated logical interfaces.
- Network Architecture: the discrete physical (network) elements of the system and the associated physical interfaces.
- Protocol Architecture: the protocol stacks involved in the operation of the system and the associated peering relationships.

collector: terrestrial components that "collect" return link transmissions from terminals and forward them towards the ground segment

control plane: plane that has a layered structure and performs the call control and connection control functions; it deals with the signalling necessary to set up, supervise and release calls and connections

repeater: terrestrial components that (mainly) repeat the satellite signal in the forward link

S-band: equivalent to 2 GHz MSS band

3.2 Abbreviations

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

ACK	Acknowledgement signalling
ACLR	Adjacent Channel Leakage power Ratio
C	number of Columns
C	Received signal (carrier) power
CDMA	Code Division Multiple Access
CGC	Complementary Ground Component
DL	Down-Link
EIRP	Effective Isotropic Radiated Power
FL	Forward-Link

GF	Gallois Field
GW	Gateway
I_0	Single-sided interference power spectral density
ML	Maximal Length
MSB	Most Significant Bit
N_0	Single-sided noise power spectral density
OVSF	Orthogonal Variable Spreading Factor
PCCC	Parallel Concatenated Convolutional Code
PCCC	Parallel Concatenated Convolutional Code
PCCH	Physical Control Channel
PDCH	Physical Data Channel
PN	Pseudo-Noise
RACH	Random Access Channel
RL	Return-Link
Rx	Receive
SAT	SSA Access Table
SCT	SSA Configuration Table
SDT	SSA Dynamic Configuration Table
SF	Spreading Factor
SRRC	Square Root Raised Cosine
SSA	Spread Spectrum Aloha
TFI	Transport Format Indication
Tx	Transmitter
UL	Up-Link
ULB	Up-Link Burst

4 General description

The present document specifies the physical layer for the Asynchronous Access option of the Return Link using the Spread Spectrum Aloha (SSA) technique [i.1] and [i.2].

The specification has similarities with 3GPP specifications [1], [2] and the differences are described.

The present specification covers the Return-Link (RL) satellite transmission and the Up-Link (UL) transmission to the terrestrial collectors. Although different configurations and parameters may apply to the RL and to UL, the radio interface is in both cases the one defined in the present document.

S-MIM asynchronous access is intended for access to interactive messaging services, since it does not require coordination between users thus minimizing the signalling overhead required for the access control. Other advantages of S-MIM asynchronous access are:

- Ability for the space segment to operate with full frequency reuse. This will contribute to further improvement of the system efficiency.
- Feasibility of band sharing with other access schemes due to the spread spectrum characteristics.

4.1 Relationship to other layers

4.1.1 General Protocol Architecture

The overall protocol architecture for the return link of S-MIM asynchronous access is shown in Figure 4.1.

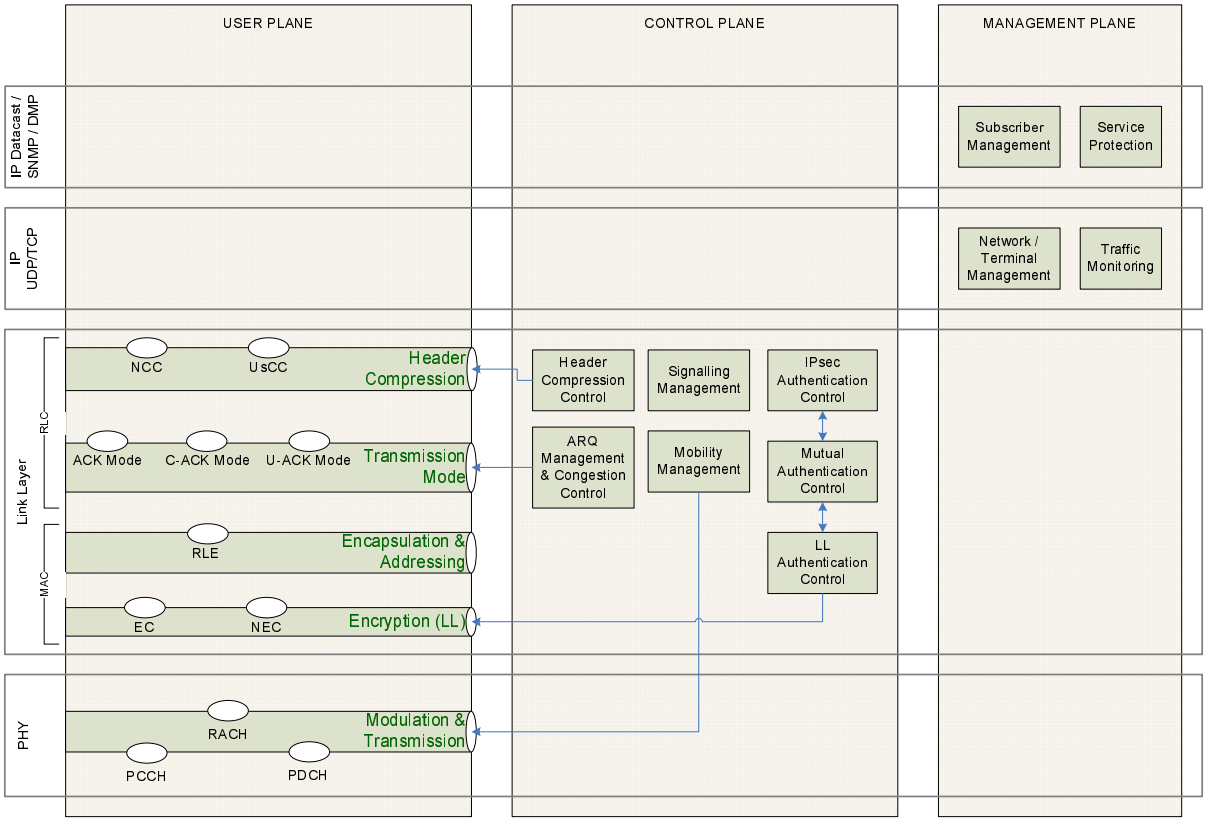

Figure 4.1: Protocol Architecture for the Asynchronous Return Link

The circles between different layer/sub-layers indicate Service Access Points (SAPs).

The physical layer provides an interface to the Medium Access Control (MAC) sub-layer via Transport channel(s). A transport channel is characterized by how and which information is transferred over the radio interface.

Physical Channels are defined in the physical layer and are characterized by the physical resources (time, frequency, code, and space) that are used to transport data/control/signalling to/from a single user or a multitude of users.

The Control Plane is concerned with all aspects of signalling, for example the signalling of the Return Link scrambling sequence to the user terminal as described in TS 102 721-6 [3].

The present document is concerned with the Physical Layer.

4.1.2 Services provided to higher layers

The physical layer offers data transport services to higher layers. The physical layer is expected to perform the following functions in order to provide the data transport service:

- Error detection on transport channel and indication to higher layers.
- FEC encoding/decoding of transport channels.
- Rate matching of coded transport channel to physical data channel.
- Mapping of coded transport channel on physical data channel.
- Power weighting and combining of physical channels.
- Modulation and spreading/demodulation and de-spreading of physical channels.
- Frequency and time (chip, bit, burst) synchronisation.
- Radio characteristics measurements and indication to higher layers (for further study).
- RF processing.

4.2 Transmitter functional architecture

In the transmission direction Physical Layer functional block diagram is shown in Figure 4.2.

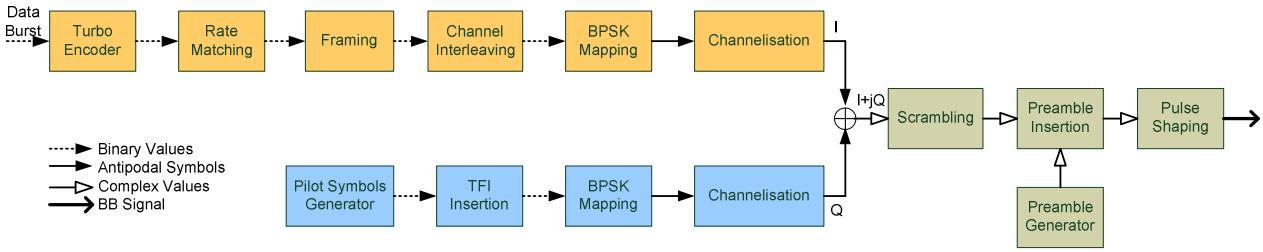

Figure 4.2: Transmitter Functional Block Diagram

4.3 Channel descriptions

4.3.1 Transport channel

There is a single transport channel: the Random Access Channel (RACH). The RACH is characterized by limited size data burst length, a collision risk and by the use of open loop power control.

Three possible maximum sizes of the RACH data burst length are supported, namely 300, 600 and 1 200 bits. Depending on the effective size of the RACH data burst length and of the size of the optional CRC, rate matching will be performed after FEC encoding.

4.3.2 Physical channels

Two physical channels are defined, namely the PDCH (Physical Data Channel) used to carry the RACH data burst and the PCCH (Physical Control Channel) used to carry physical layer signalling information. The PDCH and the PCCH are I/Q code multiplexed to form an Up-Link Burst (ULB), composed of three parts (Figure 4.3):

- a preamble
- a Physical Data Channel (PDCH), uniquely determined by its chip rate, spreading factor and data burst length as detailed in clause 5.1
- the Physical Control Channel (PCCH), uniquely determined by its chip rate as detailed in clause 5.2.

The preamble is transmitted before the start of the PDCH and PCCH.

Figure 4.3: The Up-Link Burst and its constituent parts

The PDCH will carry the RACH data burst followed by an optional CRC defined in clause 5.1.1.

The PCCH carries physical layer signalling and reference symbols to allow coherent demodulation of the PDCH channel. The physical layer signalling conveys the Transport Format Indication (TFI) to the receiving side as defined in clause 5.2.1.

For Radio Channel details see clause 8.

5 Physical Channel Structure

5.1 PDCH structure

As shown in Figure 5.1, the PDCH data burst (after channel encoding according to clause 6.1), is divided into a variable number of frames ranging from 3 (Option 1) up to 24 (Option 4). Each frame has a fixed duration of 10 ms. The PDCH length (in time or, equivalently, in frames) is indicated with the term TTI (Transmission Time Interval).

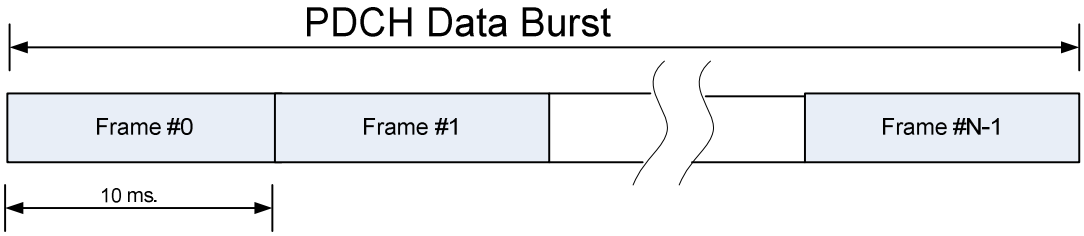

Figure 5.1: PDCH Composition

Figure 5.2 shows the structure of the 10 ms frame, which is split into 15 slots, each of length T_{slot} = 0,667 ms. Each slot consists of $10*2^k$ bits, where k=0 or 1 (k only equal to 0 for chip rate 240 kchip/s).

The set of allowed parameters of the PDCH is reported in Table 5.1. Note that the number of info bits reported in Table 5.1 is only indicative. Exploiting the rate matching mechanism, the code rate can be changed to accommodate variations in the RACH data burst length without impacting the PDCH data burst length. Clearly for not penalising system performance it is expected that the code rate is not increased significantly above the 1/3 nominal code rate.

Rate matching as specified in clause 6.2 may thus be used to adapt the code rate to the RACH data burst length, as defined in TS 102 721-6 [3].

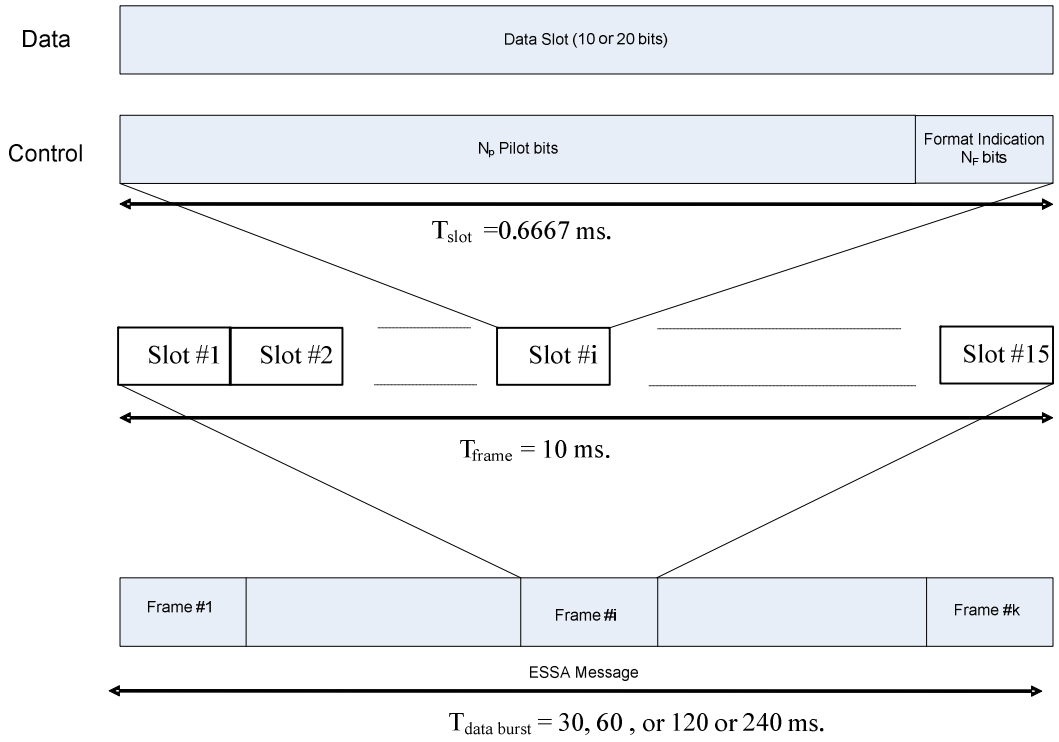

Figure 5.2: Structure of the PDCH and PCCH

Table 5.1: Allowed PDCH configurations

PDCH Configuration ID	Chip rate (kchip/s)	Ch. Bit Rate (kbps)	SF	Info Bits/ Burst	Ch. Bits/ Burst	Ch. Bits/ Frame	Ch. Bits/ Slot	Data Burst Length (Frames)
$CR_{3\,840}SF_{256}DBL_{24}$	3 840	15	256	1 200	3 600	150	10	24
$CR_{3\,840}SF_{256}DBL_{12}$	3 840	15	256	600	1 800	150	10	12
$CR_{3\,840}SF_{256}DBL_{6}$	3 840	15	256	300	900	150	10	6
$CR_{3\,840}SF_{128}DBL_{12}$	3 840	30	128	1 200	3 600	300	20	12
$CR_{3\,840}SF_{128}DBL_{6}$	3 840	30	128	600	1 800	300	20	6
$CR_{3\,840}SF_{128}DBL_{3}$	3 840	30	128	300	900	300	20	3
$CR_{1\,920}SF_{128}DBL_{24}$	1 920	15	128	1 200	3 600	150	10	24
$CR_{1\,920}SF_{128}DBL_{12}$	1 920	15	128	600	1 800	150	10	12
$CR_{1\,920}SF_{128}DBL_{6}$	1 920	15	128	300	900	150	10	6
$CR_{1\,920}SF_{64}DBL_{12}$	1 920	30	64	1 200	3 600	300	20	12
$CR_{1\,920}SF_{64}DBL_{6}$	1 920	30	64	600	1 800	300	20	6
$CR_{1\,920}SF_{64}DBL_{3}$	1 920	30	64	300	900	300	20	3
$CR_{240}SF_{16}DBL_{24}$	240	15	16	1 200	3 600	150	10	24
$CR_{240}SF_{16}DBL_{12}$	240	15	16	600	1 800	150	10	12
$CR_{240}SF_{16}DBL_{6}$	240	15	16	300	900	150	10	6

5.1.1 CRC format

The PDCH includes an optional 16 bit or 8 bit CRC. If a CRC is used the parity bits are generated by one of the following cyclic generator polynomials:

$$g_{CRC16}(D) = D^{16} + D^{12} + D^5 + 1$$

$$g_{CRC8}(D) = D^8 + D^7 + D^4 + D^3 + D + 1$$

Denote the bits in a RACH data burst delivered to layer 1 by b_1, b_2, b_3, … b_N, and the parity bits by p_1,p_2, … p_L. N is the length of message unit and L is 16, 8, or 0 depending on what is signalled from higher layers.

The encoding is performed in a systematic form, which means that in GF(2), the polynomial

$$b_1D^{N+15} + b_2D^{N+14} + \ldots + b_ND^{16} + p_1D^{15} + p_2D^{14} + \ldots + p_{15}D^1 + p_{16}$$

yields a remainder equal to 0 when divided by $g_{CRC16}(D)$. Similarly,

$$b_1D^{N+7} + b_2D^{N+6} + \ldots + b_ND^8 + p_1D^7 + p_2D^6 + \ldots + p_7D^1 + p_8$$

yields a remainder equal to 0 when divided by $g_{CRC8}(D)$.

If the CRC is used, rate matching as described in clause 6.2.2 might become necessary in order to keep the number of channel bits per burst equal to values reported in Table 5.1. The use (and length) of the CRC is broadcast in the SCT table in TS 102 721-6 [3] for each possible TFI.

CRC shift register shall be initialised to 0x0000.

NOTE: For operation with interference cancellation (which is the baseline for SSA, at least when used on satellite links) CRC is to be regarded as mandatory in practice. In absence of CRC the turbo decoder should be used to provide a likelihood of the correct decoding. Given the uncertainty about the quality of such indication it appears safer to rely always on CRC when interference cancellation is considered.

5.2 PCCH structure

The PCCH consists of N_p known pilot bits to support channel estimation for coherent detection and N_F bits for the TFI (Transport Format Indication) message per slot. The size of each PCCH slot is always 10 bits, as shown in Table 5.2.

The TFI carries information associated with the packet format and packet length. The value of N_p is broadcast in the SCT table in TS 102 721-6 [3]. Its complement to ten are the TFI bits per slot.

Table 5.2: Allowed PCCH configurations

PCCH Configuration ID	Chip rate (kchip/s)	Symbol Rate (kbaud)	SF	Symbols/ Frame	Symbols/ Slot	Data Burst Length (Frames)	N_F
$CR_{3\,840}$	3 840	15	256	150	10	depends on PDCH	$10\text{-}N_p$
$CR_{1\,920}$	1 920	15	128	150	10		
CR_{240}	240	15	16	150	10		

5.2.1 Pilot symbols format

Pilot symbols transmitted on the PCCH are obtained through a ML PN generator.

The scrambling sequence is a segment of an ML sequence defined by polynomial $1+X^4+X^9$. The initial loading of the shift register shall be equal to 101000000 as indicated in Figure 5.3. The shift register shall be reset at the initial loading at the start of each PDCH data burst.

The pilot symbol generator is clocked in correspondence to every pilot symbols in PCCH. The generated sequence corresponds to the pilot pattern of the PCCH.

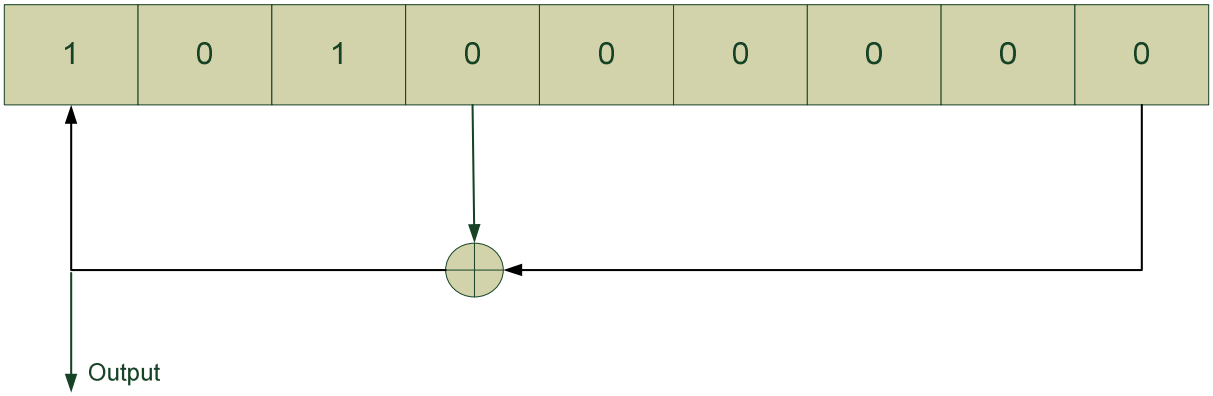

Figure 5.3: Pilot symbols generation

5.2.2 Transport Format Indication format

The TFI message carries 5 information bits which are thus able to address up to 32 different packet structures.

This message is encoded in a (15, 5) codeword obtained extending an ML code (15, 4) generated by the primitive polynomial $h(x)=x^4+x+1$ with the antipodal codewords.

NOTE: The generator polynomial is thus $g(x)=(x^{15}-1)/h(x)=x^{11}+x^8+x^7+x^5+x^3+x^2+x+1$.

The mapping of info bit field to codeword is:

- The MSB of the Format message determine if the set of antipodal codewords has to be selected (a 1 select the antipodal set).
- The remaining four bits select the zero codeword (or its antipodal) in case they are all zero, or the ML sequence generated by the given primitive polynomial when the shift register is initialized with the four info bits (Figure 5.4).

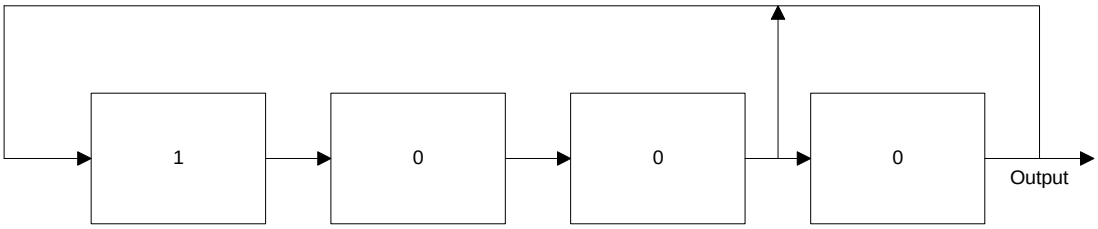

Figure 5.4: ML code generation

The coding procedure is performed independently for each frame. As each frame contains 15 slots, one bit per slot is thus generated. If more than one bit per slot is allocated to the TFI (i.e. $N_F > 1$), the bit is repeated multiple times as required.

The entire codeword is then repeated within each frame of the PDCH data burst. As the minimum PDCH data burst size is 3 frames, a terminal may softly combine bits over three frames before decoding.

5.3 Transport-to-Physical Channel Mapping

The mapping between the transport channel defined in clause 4.3.1 and the physical channels defined in clause 4.3.2 is unambiguously determined through Table 5.1 depending on the chip rate, the spreading factor and the required PDCH data burst length (or equivalently, the required info bits per burst). These parameters are selected by the system operator and broadcast in the SCT table in TS 102 721-6 [3].

6 Channel Coding and Interleaving

6.1 Channel Coding

The Turbo-coder has a nominal code rate of 1/3, but may slightly differ from 1/3 owing to the rate matching capability as defined in clause 6.2.2. The following turbo-code specifications are taken from the 3GPP [1] specifications.

The scheme of Turbo coder is a Parallel Concatenated Convolutional Code (PCCC) with two 8-state constituent encoders and one Turbo code internal interleaver. The coding rate of Turbo coder is 1/3. The structure of Turbo coder is illustrated in Figure 6.1.

6.1.1 Constituent code

The transfer function of the 8-state constituent code for PCCC is:

$$G(D) = \left[1, \frac{g_1(D)}{g_0(D)}\right],$$

where

$$g_0(D) = 1 + D^2 + D^3,$$

$$g_1(D) = 1 + D + D^3.$$

The initial value of the shift registers of the 8-state constituent encoders shall be all zeros when starting to encode the input bits.

Output from the Turbo coder is:

$$x_1, z_1, z'_1, x_2, z_2, z'_2, \ldots, x_K, z_K, z'_K,$$

where $x_1, x_2, \ldots, x_K$ are the bits input to the Turbo coder i.e. both first 8-state constituent encoder and Turbo code internal interleaver, and K is the number of bits, and $z_1, z_2, \ldots, z_K$ and $z'_1, z'_2, \ldots, z'_K$ are the bits output from first and second 8-state constituent encoders, respectively.

The bits output from Turbo code internal interleaver are denoted by $x'_1, x'_2, \ldots, x'_K$, and these bits are to be input to the second 8-state constituent encoder.

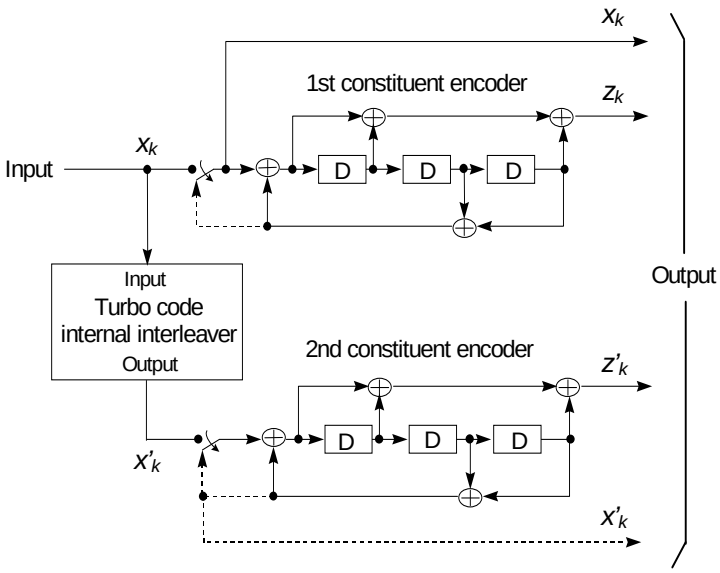

Figure 6.1: Structure of rate 1/3 Turbo coder (dotted lines apply for trellis termination only)

6.1.2 Trellis termination for turbo coder

Trellis termination is performed by taking the tail bits from the shift register feedback after all information bits are encoded. Tail bits are padded after the encoding of information bits.

The first three tail bits shall be used to terminate the first constituent encoder (upper switch of Figure 6.1 in lower position) while the second constituent encoder is disabled. The last three tail bits shall be used to terminate the second constituent encoder (lower switch of Figure 6.1 in lower position) while the first constituent encoder is disabled.

The transmitted bits for trellis termination shall then be:

$$x_{K+1}, z_{K+1}, x_{K+2}, z_{K+2}, x_{K+3}, z_{K+3}, x'_{K+1}, z'_{K+1}, x'_{K+2}, z'_{K+2}, x'_{K+3}, z'_{K+3}.$$

6.1.3 Turbo code internal interleaver

The Turbo code internal interleaver consists of bits-input to a rectangular matrix, intra-row and inter-row permutations of the rectangular matrix, and bits-output from the rectangular matrix with pruning. The bits input to the Turbo code internal interleaver are denoted by $x_1, x_2, x_3, \ldots, x_K$, where K is the integer number of the bits and takes one value of $40 \leq K \leq 5\,114$. The relation between the bits input to the Turbo code internal interleaver and the bits input to the channel coding is defined by $x_k = o_{irk}$ and $K = K_i$.

The following specific symbols are used in clauses 6.1.3.1 to 6.2.1:

K Number of bits input to Turbo code internal interleaver

R Number of rows of rectangular matrix

C Number of columns of rectangular matrix

p Prime number

v Primitive root

s(i) Base sequence for intra-row permutation

qj Minimum prime integers

rj Permuted prime integers

T(j) Inter-row permutation pattern

Uj(i) Intra-row permutation pattern

i Index of matrix

j Index of matrix

k Index of bit sequence

6.1.3.1 Bits-input to rectangular matrix

The bit sequence input to the Turbo code internal interleaver x_k is written into the rectangular matrix as follows.

(1) Determine the number of rows *R* of the rectangular matrix such that:

$$R = \begin{cases} 5, \text{if } (40 \le K \le 159) \\ 10, \text{if } ((160 \le K \le 200) \text{ or } (481 \le K \le 530)) \\ 20, \text{if } (K = \text{any other value}) \end{cases}$$

where the rows of rectangular matrix are numbered 0, 1, 2, …, *R* - 1 from top to bottom.

(2) Determine the number of columns *C* of rectangular matrix such that:

if (481 ≤ *K* ≤ 530) then

p = *53* and *C* = *p*.

else

Find minimum prime *p* such that

(*p* + 1) - *K*/*R* ≥ 0,

and determine *C* such that

if (*p* - *K*/*R* ≥ 0) then

if (*p* - 1 - *K*/*R* ≥ 0) then

C = *p* - 1.

else

C = *p*.

end if

else

C = *p* + 1

end if

end if

where the columns of rectangular matrix are numbered 0, 1, 2, …, *C* - 1 from left to right.

(3) Write the input bit sequence x_k into the $R \times C$ rectangular matrix row by row starting with bit x_1 in column 0 of row 0:

$$\begin{bmatrix} x_1 & x_2 & x_3 & \cdots x_C \\ x_{(C+1)} & x_{(C+2)} & x_{(C+3)} & \cdots x_{2C} \\ \vdots & \vdots & \vdots & \cdots \vdots \\ x_{((R-1)C+1)} & x_{((R-1)C+2)} & x_{((R-1)C+3)} & \cdots x_{RC} \end{bmatrix}$$

6.2 Intra-row and inter-row permutations

After the bits-input to the $R \times C$ rectangular matrix, the intra-row and inter-row permutations for the $R \times C$ rectangular matrix are performed by using the following algorithm.

(1) Select a primitive root v from Table 6.1.

(2) Construct the base sequence $s(i)$ for intra-row permutation as:

$s(i) = [v \times s(i - 1)] \bmod p$, $i = 1, 2, \ldots, (p - 2)$., and $s(0) = 1$.

(3) Let $q_0 = 1$ be the first prime integer in $\{q_j\}$, and select the consecutive minimum prime integers $\{q_j\}$ ($j = 1, 2, \ldots, R - 1$) such that:

g.c.d. $\{q_j, p - 1\} = 1$, $q_j > 6$, and $q_j > q_{(j-1)}$,

where g.c.d. is greatest common divisor.

(4) Permute $\{q_j\}$ to make $\{r_j\}$ such that

$r_{T(j)} = q_j$, $j = 0, 1, \ldots, R - 1$,

where $T(j)$ ($j = 0, 1, 2, \ldots, R - 1$) is the inter-row permutation pattern defined as the one of the following four kind of patterns: Pat_1, Pat_2, Pat_3 and Pat_4 depending on the number of input bits K.

$$\{T(0), T(1), T(2), \ldots, T(R-1)\} = \begin{cases} Pat_4 & \text{if } (40 \le K \le 159) \\ Pat_3 & \text{if } (160 \le K \le 200) \\ Pat_1 & \text{if } (201 \le K \le 480) \\ Pat_3 & \text{if } (481 \le K \le 530) \\ Pat_1 & \text{if } (531 \le K \le 2280) \\ Pat_2 & \text{if } (2281 \le K \le 2480) \\ Pat_1 & \text{if } (2481 \le K \le 3160) \\ Pat_2 & \text{if } (3161 \le K \le 3210) \\ Pat_1 & \text{if } (3211 \le K \le 5114) \end{cases},$$

where Pat_1, Pat_2, Pat_3 and Pat_4 have the following patterns respectively.

Pat_1: {19, 9, 14, 4, 0, 2, 5, 7, 12, 18, 10, 8, 13, 17, 3, 1, 16, 6, 15, 11}

Pat_2: {19, 9, 14, 4, 0, 2, 5, 7, 12, 18, 16, 13, 17, 15, 3, 1, 6, 11, 8, 10}

Pat_3: {9, 8, 7, 6, 5, 4, 3, 2, 1, 0}

Pat_4: {4, 3, 2, 1, 0}

(5) Perform the j-th ($j = 0, 1, 2, \ldots, R - 1$) intra-row permutation as:

if ($C = p$) then

$U_j(i) = s([i \times r_j] \bmod (p - 1))$, $i = 0, 1, 2, \ldots, (p - 2)$., and $U_j(p - 1) = 0$,

where $U_j(i)$ is the input bit position of i-th output after the permutation of j-th row.

end if

if ($C = p + 1$) then

$U_j(i) = s([i \times r_j] \bmod (p - 1))$, $i = 0, 1, 2, \ldots, (p - 2)$., $U_j(p - 1) = 0$, and $U_j(p) = p$,

where $U_j(i)$ is the input bit position of i-th output after the permutation of j-th row, and

if ($K = C \times R$) then

Exchange $U_{R-1}(p)$ with $U_{R-1}(0)$.

end if

end if

if ($C = p - 1$) then

$U_j(i) = s([i \times r_j] \bmod (p - 1)) - 1$, $i = 0, 1, 2, \ldots, (p - 2)$,

where $U_j(i)$ is the input bit position of i-th output after the permutation of j-th row.

end if

(6) Perform the inter-row permutation based on the pattern $T(j)$ ($j = 0, 1, 2, \ldots, R - 1$),

where $T(j)$ is the original row position of the j-th permuted row.

Table 6.1: Table of prime *p* and associated primitive root *v*

p	v	p	v	p	v	p	v	p	v
7	3	47	5	101	2	157	5	223	3
11	2	53	2	103	5	163	2	227	2
13	2	59	2	107	2	167	5	229	6
17	3	61	2	109	6	173	2	233	3
19	2	67	2	113	3	179	2	239	7
23	5	71	7	127	3	181	2	241	7
29	2	73	5	131	2	191	19	251	6
31	3	79	3	137	3	193	5	257	3
37	2	83	2	139	2	197	2		
41	6	89	3	149	2	199	3		
43	3	97	5	151	6	211	2		

6.2.1 Bits-output from rectangular matrix with pruning

After intra-row and inter-row permutations, the bits of the permuted rectangular matrix are denoted by y'_k:

$$\begin{bmatrix} y'_1 & y'_{(R+1)} & y'_{(2R+1)} & \cdots & y'_{((C-1)R+1)} \\ y'_2 & y'_{(R+2)} & y'_{(2R+2)} & \cdots & y'_{((C-1)R+2)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ y'_R & y'_{2R} & y'_{3R} & \cdots & y'_{CR} \end{bmatrix}$$

The output of the Turbo code internal interleaver is the bit sequence read out column by column from the intra-row and inter-row permuted $R \times C$ matrix starting with bit y'_1 in row 0 of column 0 and ending with bit y'_{CR} in row R - 1 of column C - 1. The output is pruned by deleting bits that were not present in the input bit sequence, i.e. bits y'_k that corresponds to bits x_k with $k > K$ are removed from the output. The bits output from Turbo code internal interleaver are denoted by $x'_1, x'_2, \ldots, x'_K$, where x'_1 corresponds to the bit y'_k with smallest index k after pruning, x'_2 to the bit y'_k with second smallest index k after pruning, and so on. The number of bits output from Turbo code internal interleaver is K and the total number of pruned bits is:

$$R \times C - K.$$

6.2.2 Rate matching

The rate matching algorithm described below is used to adapt the number of bits at the output of the turbo encoder (which is three times the input block length in bits plus 12 bits for the encoder tails) to the available channel bits. If the CRC (see clause 5.1.1) is used, rate matching might become necessary in order to keep the number of info bits equal to values reported in Table 5.1 and thus ensuring consistency with the transport channel definition (see clause 4.3.1).

The systematic bits of turbo encoded codewords shall not be punctured, the other bits may be punctured. The systematic bits, first parity bits, and second parity bits in the bit sequence input to the rate matching block are therefore separated into three sequences if puncturing is required (see Figure 6.2).

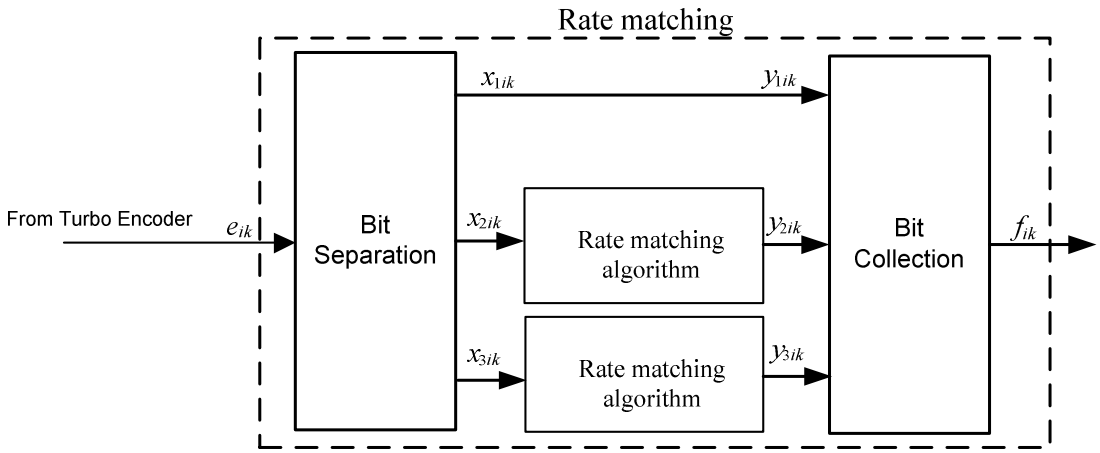

Figure 6.2: Rate Matching in case of puncturing

The first sequence contains:

- All of the systematic bits from the turbo encoded block.
- Some of the systematic, first parity and second parity bits that are for trellis termination.

The second sequence contains:

- All of the first parity bits from the turbo encoded block.
- Some of the systematic, first parity and second parity bits that are for trellis termination.

The third sequence contains:

- All of the second parity bits from the turbo encoded block.
- Some of the systematic, first parity and second parity bits that are for trellis termination.

Puncturing is applied only to the second and third sequence. Bit Collection is the inverse operation than Bit Separation. After Bit Collection the bits marked for puncturing by the rate matching algorithm are removed.

In case repetition is required instead of puncturing no bit separation / collection is done (Figure 6.3).

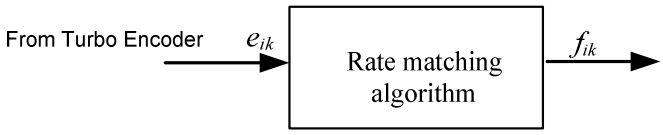

Figure 6.3: Rate Matching in case of repetition

The Rate Matching algorithm operates as follows:

- Let *B* and *A* respectively be the number of available bits before puncturing or repetition and the number of bits which fits the system requirements (i.e. the number of bits after the puncturing or repetition). Clearly no puncturing or repetition is needed if *A*=*B*. If *B*>*A* a puncturing shall be performed, whilst repetition is to be done if *A*>*B*.
- Let *div*= Max Common Divisor (abs(*A*-*B*), *B*) where abs stands for absolute value.
- Then the period of the repetition or puncturing pattern is $P = B / div$.
- The number of bits to be punctured or repeated in such a period is:

  x= abs (*A*-*B*) /*div*;

To determine the puncturing pattern or repetition, i.e. a binary vector of size P containing 1 in correspondence of the bits to be punctured or repeated the following pseudo-C algorithm can be used, where *pattern* is the binary vector to be computed.

```
control=x/P;
counter=0;
for (i=0; i<P; i++)
{
    if((i*control-counter)>0)
    {
        pattern[i]=1;
        counter++;
    }
    else     pattern[i]=0;
}
if((x>counter)     pattern[0]=1;
```

6.3 Channel Interleaving

The channel interleaver is specified as the cascade of two separate channel interleavers. This allows to have the second interleaver always operating on a single frame. The first inteleaver will then take care of dispersing bits on the full PDCH data burst.

The 1st interleaver can be implemented as a block interleaver with inter-column permutations.

The output bit sequence is derived as follows:

(1) Select the number of columns C_I from Table 6.2.

(2) Determine the number of rows R_I defined as

$$R_I = X_i/C_I$$

(3) Write the input bit sequence into the $R_I \times C_I$ rectangular matrix row by row

(4) Perform the inter-column permutation based on the pattern $\{P_1 (j)\}$ (j=0,1, ..., C-1) shown in the Table 6.2.

(5) Read the output bit sequence column by column from the inter-column permuted $R_I \times C_I$ matrix.

Table 6.2: 1st Channel Interleaver

TTI	Number of columns C_1	Inter-column permutation patterns
3 frames	3	{0,2,1}
6 frames	6	{0,4,2,1,5,3}
12 frames	12	{0, 8, 4, 2, 10, 6, 1,9,5,3,11,7}
24 frames	24	{0, 16, 8, 4, 20, 12, 2,18,10,6,22,14,1,17,9,5,21,13,3,19,11,7,23,15,}

The second channel interleaver acts over a single frame. It consists of two stages. In the first stage, the input sequence is written into a rectangular matrix row by row. The second stage is inter-column permutation. The two-stage operations are described as follows, the input block length is assumed to be K_2.

First Stage:

(1) Set a column number $C_2 = 30$.

(2) Determine a row number R_2 by finding minimum integer R_2 such that:

$$K_2 \leq R_2 \times C_2.$$

(3) The input sequence of the 2nd interleaving is written into the $R_2 \times C_2$ rectangular matrix row by row.

Second Stage:

(1) Perform the inter-column permutation based on the pattern $\{P_2(j)\}$ (j=0,1, ..., C-1) that is shown in Table 6.3, where $P_2(j)$ is the original column position of the j-th permuted column.

(2) The output of the 2nd interleaving is the sequence read out column by column from the inter-column permuted $R_2 \times C_2$ matrix and the output is pruned by deleting the non-existence bits in the input sequence, where the deleting bits number l_2 is defined as:

$$l_2 = R_2 \times C_2 - K_2.$$

Table 6.3: 2nd Interleaving column permutation

Number of column C_2	Inter-column permutation pattern
30	{0, 20, 10, 5, 15, 25, 3, 13, 23, 8, 18, 28, 1, 11, 21, 6, 16, 26, 4, 14, 24, 19, 9, 29, 12, 2, 7, 22, 27, 17}

7 Spreading and Modulation

7.1 PDCH and PCCH spreading

Figure 7.1 illustrates the spreading and scrambling of the PDCH and PCCH.

Spreading is applied after mapping the control and data binary values 0 and 1 to +1 and -1 respectively and before pulse shaping. It consists of two operations:

1) the channelization operation, which transforms every symbol into a number of chips, thus increasing the bandwidth of the signal.

2) the scrambling operation, where a scrambling code is applied to the spread signal obtained by I/Q multiplexing of the data and control branches.

With the channelization operation, data symbols in the so-called I- and Q-branches are independently multiplied with an OVSF code. With the scrambling operation, the resultant signals on the I- and Q-branches are further multiplied by complex-valued scrambling code, where I and Q denote real and imaginary parts, respectively.

The real-valued signals of the I- and Q-branches are then summed and treated as a complex signal. This complex signal is then scrambled by the complex-valued scrambling code, C_{scramb}. The control channel may be adjusted by the gain factor $\beta \leq 1$.

The values of β are quantized into 4 bits, and the quantization steps are given in Table 7.1.

Table 7.1: Quantization of the gain parameters.

Signalling values for β	Quantized amplitude ratios (β)
15	1,0
14	0,9333
13	0,8666
12	0,8000
11	0,7333
10	0,6667
9	0,6000
8	0,5333
7	0,4667
6	0,4000
5	0,3333
4	0,2667
3	0,2000
2	0,1333
1	0,0667
0	Switch off

The channelization code to be used for the PCCH and for the PDCH are signalled in the SCT table in TS 102 721-[3].

NOTE 1: For the sake of clarity, the average power of the preamble and of the data + control channel part should always be the same. However for the prambles the amplitude of both I and Q component is equal, whereas for the data + control channel part the amplitude ratio should be PDCH:PCCH=1:ß

NOTE 2: Such signalling could be eliminated if fixed sequences are specified i.e. the all 1 sequences for the PCCH and the {1, -1, 1, -1, …} sequence for the PDCH).

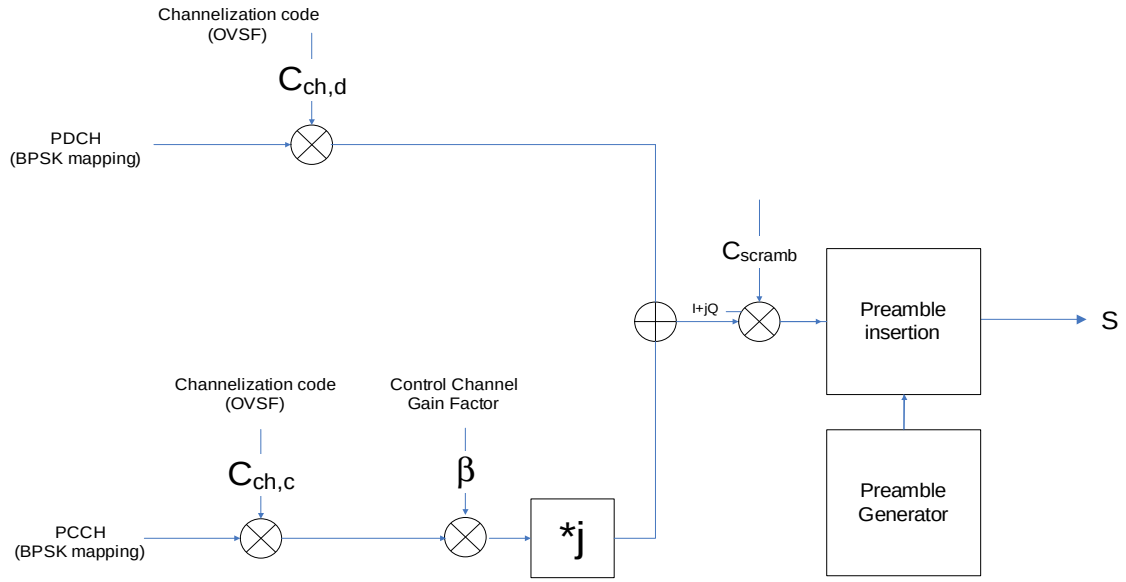

Figure 7.1: Spreading, scrambling and preamble insertion for PDCH and PCCH

7.1.1 PDCH and PCCH channelization

The PCCH is spread by the channelization code $C_{ch,c}$ and the PDCH is spread by a different channelization codes, $C_{ch,d}$. Symbols of both channels are BPSK-modulated.

The channelisation codes are Orthogonal Variable Spreading Factor (OVSF) codes that preserve the orthogonality between different physical channels (dedicated physical channels only). The OVSF codes can be defined using the code tree of the Figure 7.2.

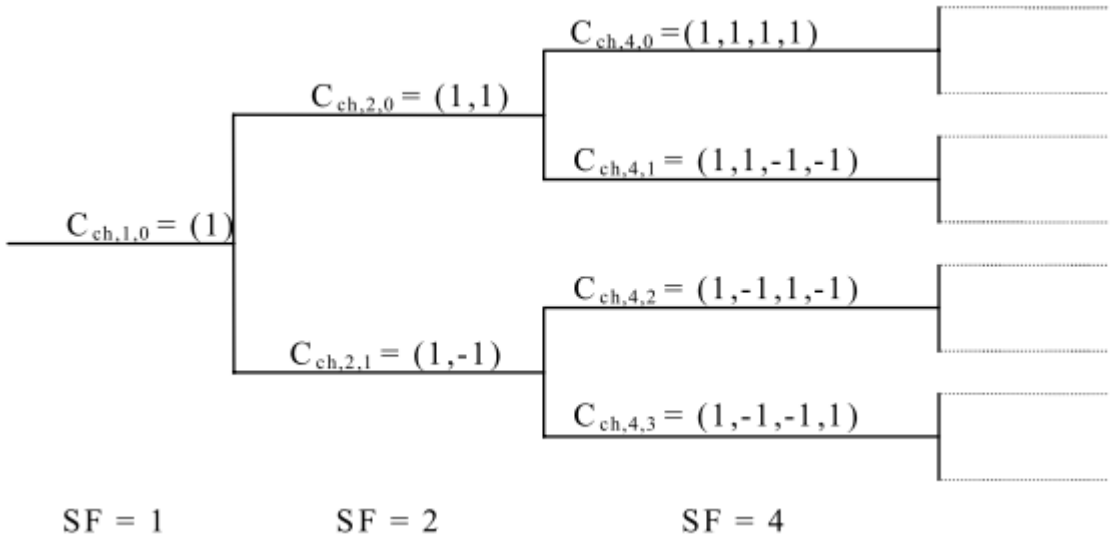

Figure 7.2: Code-tree for generation of Orthogonal Variable Spreading Factor (OVSF) codes

Channelisation codes are uniquely described as $C_{ch,SF,k}$, where SF is the spreading factor of the code and k is the code number, $0 \leq k \leq SF-1$.

Each level in the code tree defines channelisation codes of the length SF, corresponding to a spreading factor of SF in Figure 7.2.

The generation method for the channelisation code is defined as:

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

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

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

The leftmost value in each channelisation code word corresponds to the chip transmitted first in time.

7.1.2 PDCH and PCCH scrambling

The same scrambling solution as foreseen in 3GPP WCDMA is foreseen (long code option of 3GPP) [2]. However, differently from 3GPP the scrambling code period is not limited to 38 400 chips. In fact a period equal to the entire PDCH burst length (see clause 5.1) is selected to minimize code collision probability between random access users. There are 2^{24} scrambling codes. The codes are complex valued and are formed as follows: $C_{scramb} = c_1(w_0 + jc_2'w_1)$

where w_0 and w_1 are chip rate sequences defined as repetitions of:

$w_0 = \{1, 1\}$

$w_1 = \{1, -1\}$

Also, c_1 is a real chip rate code, and c_2' is a decimated version of the real chip rate code c_2.

With a decimation factor 2, c_2' is given as:

$c_2'(2k) = c_2'(2k+1) = c_2(2k)$, k=0, 1, 2….

The sequences c_1 and c_2 are constructed as the position wise modulo 2 sum of segments of two binary *m*-sequences generated by means of two generator polynomials of degree 25. Let *x*, and *y* be the two *m*-sequences respectively. The *x* sequence is constructed using the primitive (over GF(2)) polynomial $X^{25}+X^3+1$. The *y* sequence is constructed using the polynomial $X^{25}+X^3+X^2+X+1$. The resulting sequences thus constitute segments of a set of Gold sequences.The code, $c_{2,n}$ used in generating the quadrature component of the complex spreading code is a 16,777,232 chip shifted version of the code, $c_{1,n}$ used in generating the in phase component.

Code construction for $c_{1,n}$ and $c_{2,n}$ is shown in Figure 7.3.

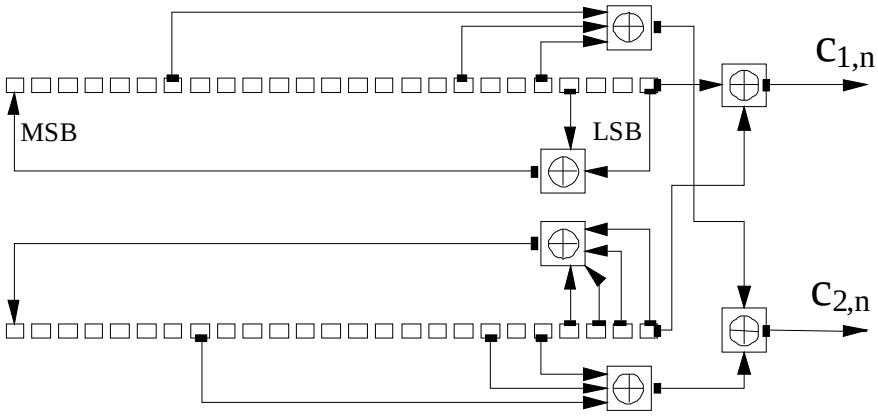

Figure 7.3: Configuration of uplink scrambling code generator

Different scrambling code can be used in the system. In practice allowing different signatures s_1 for the preamble is equivalent to use multiple scrambling/spreading codes for packets of a given beam.

7.2 Up-Link Burst Preamble

The preamble processing is illustrated in Figure 7.1.

The ULB complex quaternary preamble p is composed as follows:

$p(k)= s_1(k \text{ div } N_c)\ s_2(k \bmod N_c)$ $\quad k=0, 1, 2, \ldots P\cdot N_c-1$

where the 'div' operator is the integer division and the 'mod' operator is the modulus operator 8 i.e. the rest of an integer division).

The number of symbols, P, is 96.

The value of N_c depends on the chip rate, i.e.:

- N_c=256 for the 3,84 Mchip/s option {correct comma and dot decimal separator in all documents}
- N_c=128 for the 1,92 Mchip/s option
- N_c=16 for the 240 kchip/s option

The preamble can thus be thought as composed of a sequence s_1 of length P=96 symbols and symbol rate 15 ksymbol/s spread by a PN code s_2 of period N_c and spreading factor N_c.

7.2.1 Sequence s_1

The sequence s_1 is a quaternary sequence with period equal to 96 constructed by combining two real binary sequences into a complex one.

Each of the two real sequences is constructed as the position-wise modulo 2 sum of *P* chip segments of two binary *m*-sequences generated by means of two generator polynomials of degree 9. The resulting sequences thus constitute segments of a set of Gold sequences.

Let *x* and *y* be the two binary m-sequences respectively. The *x* sequence is constructed using the primitive (over GF(2)) polynomial $1+X^4+X^9$. The y sequence is constructed using the polynomial $1+X+X^3+X^4+X^9$.

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

The *m*-sequences *x* and *y* are constructed as:

Initial conditions:

x (0)=1, x(1)= x(2)=...= x (8)= x (9)=0

y(0)=y(1)= … =y(9)= y(9)=1

Recursive definition of subsequent symbols:

x(i+9) =x(i+4) + x(i) modulo 2,

y(i+9) = y(i+4)+y(i+3)+y(i+1)+y(i) modulo 2.

The n:th Gold code sequence z_n, $n=0,1,2,\ldots,2^9-2$, is then defined as:

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

These binary code words are converted to real valued sequences by the transformation '0' -> '+1', '1' -> '-1'.

Finally, the n:th complex scrambling code sequence $s_{1,n}$ is defined as (the lowest index corresponding to the chip scrambled first in each frame):

$$s_{1,n}(i) = z_n(i) + j\, z_n(i+256) \qquad i=0,1,\ldots,P-1.$$

7.2.2 Sequence s_2

The sequence s_2 is of length 256, or 128 or 16 depending on the PDCH chip rate (3,94 Mchip/s, or 1,92 Mchip/s or 240 kchip/s respectively.

We will assume that only a single s_2 sequence is used in a given beam. If multiple preambles are required in a given beam to reduce the collision probability then this may be obtained by using multiple s_1 signatures.

The sequence s_2 reported below for the different spreading factors are quaternary *Golay* sequences generated with the algorithm specified in Annex A. The actual sequence to be used is signalled in the SCT table in TS 102 721-6 [3].

7.2.2.1 SF = 16

The two sequences are obtained with weight [+j; +j; +j ; +j] and permutation [2; 1; 8; 4;]:

Sequence 1 is: [1, j, j ,1, j, -1, -1, j, j, 1, -1, -j, 1, -j, j, -1], * exp(j π/4)

Sequence 2 is: [1, j, -j, -1, j, -1, 1, -j, j, 1, 1, j, 1, -j, -j, 1] * exp(j π/4)

7.2.2.2 SF = 128

The two sequences are obtained with weight [+j; -1; +j; -1; j; -1; -1] and permutation [16; 1; 64; 4; 8; 2; 32]:

Sequence 1 is:

```
[+1, -1, -1, +1, -1, +1 ,+1 ,-1, +j, -j, +j, -j, +j, -j, +j, -j,
-1, +1, -1, +1, +1, -1, +1, -1, -j, +j, +j, -j, -j, +j, +j, -j,
+j, +j, -j, -j, +j, +j, -j, -j, -1, -1, -1, -1, +1, +1, +1, +1,
-j, -j, -j, -j, -j, -j, -j, -j, +1, +1, -1, -1, -1, -1, +1, +1,
+j, +j, -j, -j, -j, -j, +j, +j, -1, -1, -1, -1, -1, -1, -1, -1,
-j, -j, -j, -j, +j, +j, +j, +j, +1, +1, -1, -1, +1, +1, -1, -1,
-1, +1, +1, -1, -1, +1, +1, -1, -j, +j, -j, +j, +j, -j, +j, -j,
+1, -1, +1, -1, +1, -1, +1, -1, +j, -j, -j, +j, -j, +j,+ j, -j] *exp(jπ/4)
```

Sequence 2 is:

```
[+1, -1, -1, +1, -1, +1, +1, -1, +j, -j, +j, -j, +j, -j, +j, -j,
-1, +1,-+1, +1, +1, -1, +1, -1, -j, +j, +j, -j, -j, +j, +j, -j,
+j, +j, -j, -j, +j, +j, -j, -j, -1, -1, -1, -1, +1, +1, +1, +1,
-j, -j, -j, -j, -j, -j, -j, -j, +1, +1, -1, -1, -1, -1, +1, +1,
-j, -j, +j, +j, +j, +j, -j, -j, +1, +1, +1, +1, +1, +1, +1, +1,
+j, +j, +j, +j, -j, -j, -j, -j, -1, -1, +1, +1, -1, -1, +1, +1,
 +1, -1, -1, +1, +1, -1, -1, +1, +j, -j, +j, -j, -j, +j, -j, +j,
 -1, +1, -1, +1, -1, +1, -1, +1, -j, +j, +j, -j, +j, -j, -j, +j] *exp(jπ/4)
```

7.2.2.3 SF = 256

The two sequences are obtained with weight [+j; +j; +j; -1; +j; +j; -1] and permutation [16; 8; 4; 32; 2; 64; 1; 128;]:

Sequence 1 is:

```
[+1, +j, -1, -j, +j, -1, -j,+1, +j, -1, -j,+1,+1, +j, -1, -j, +j, -1, -j,+1, -1,
-j,+1, +j,+1, +j, -1, -j, -j,+1, +j, -1, +j, -1, +j, -1,+1, +j,+1, +j, -1, -j, -
1, -j, -j,+1, -j,+1, -1, -j, -1, -j, +j, -1, +j, -1, +j, -1, +j, -1, -1, -j, -1,
-j, +j,+1, +j,+1, -1, +j, -1, +j, -1, +j, -1, +j, +j,+1, +j,+1, -1, +j, -1, +j,
-j, -1, -j, -1, +j,+1, +j,+1,+1, -j,+1, -j, -1, +j,+1, -j, +j,+1, -j, -1, -j, -
1, +j,+1,+1, -j, -1, +j, -j, -1, +j,+1, -1, +j,+1, -j, -1, +j,+1, -j, -j, -1,
+j,+1, -1, +j,+1, -j, -j, -1, +j,+1, -j, -1, +j,+1, -1, +j,+1, -j, -j, -1,
+j,+1,+1, -j, -1, +j, -1, +j,+1, -j, +j,+1, -j, -1, -j, -1, -j, -1, -1, +j, -1,
+j,+1, -j,+1, -j, +j,+1, +j,+1,+1, -j,+1, -j, -j, -1, -j, -1, -j, -1, -j, -1,+1,
-j,+1, -j, -j,+1, -j,+1,+1, +j,+1, +j,+1, +j,+1, +j, -j,+1, -j,+1,+1, +j,+1, +j,
+j, -1, +j, -1, -j,+1, -j,+1, -1, -j, -1, -j,+1, +j, -1, -j, -j,+1, +j, -1, +j,
-1, -j,+1, -1, -j,+1, +j, +j, -1, -j,+1,+1, +j, -1, -j,+1, +j, -1, -j, +j, -1, -
j,+1] *exp(j π/4)
```

Sequence 2 is:

```
[+1, +j, -1, -j, +j, -1, -j, +1, +j, -1, -j, +1, +1, +j, -1, -j,
 -j, +1, +j, -1, +1, +j, -1, -j, -1, -j, +1, +j, +j, -1, -j, +1,
 +j, -1, +j, -1, +1, +j, +1, +j, -1, -j, -1, -j, -j, +1, -j, +1,
 +1, +j, +1, +j, -j, +1, -j, +1, -j, +1, -j, +1, +1, +j, +1, +j,
 +j, +1, +j, +1, -1, +j, -1, +j, -1, +j, -1, +j, +j, +1, +j, +1,
 +1, -j, +1, -j, +j, +1, +j, +1, -j, -1, -j, -1, -1, +j, -1, +j,
 -1, +j, +1, -j, +j, +1, -j, -1, -j, -1, +j, +1, +1, -j, -1, +j,
 +j, +1, -j, -1, +1, -j, -1, +j, +1, -j, -1, +j, +j, +1, -j, -1,
 -1, +j, +1, -j, -j, -1, +j, +1, -j, -1, +j, +1, -1, +j, +1, -j,
 +j, +1, -j, -1, -1, +j, +1, -j, +1, -j, -1, +j, -j, -1, +j, +1,
 -j, -1, -j, -1, -1, +j, -1, +j, +1, -j, +1, -j, +j, +1, +j, +1,
 -1, +j, -1, +j, +j, +1, +j, +1, +j, +1, +j, +1, -1, +j, -1, +j,
 -j, +1, -j, +1, +1, +j, +1, +j, +1, +j, +1, +j, -j,+1, -j, +1,
 -1, -j, -1, -j, -j, +1, -j, +1, +j, -1, +j, -1, +1, +j, +1, +j,
 +1, +j, -1, -j, -j, +1, +j, -1, +j, -1, -j, +1, -1, -j, +1, +j,
 -j, +1, +j, -1, -1, -j, +1, +j, -1, -j, +1, +j, -j, +1, +j, -1] *exp(jπ/4)
```

7.3 Modulation

Modulation of the complex-valued chip sequence S generated by the spreading process described in clause 7.1 is shown below in Figure 7.4 for a given uplink frequency ω.

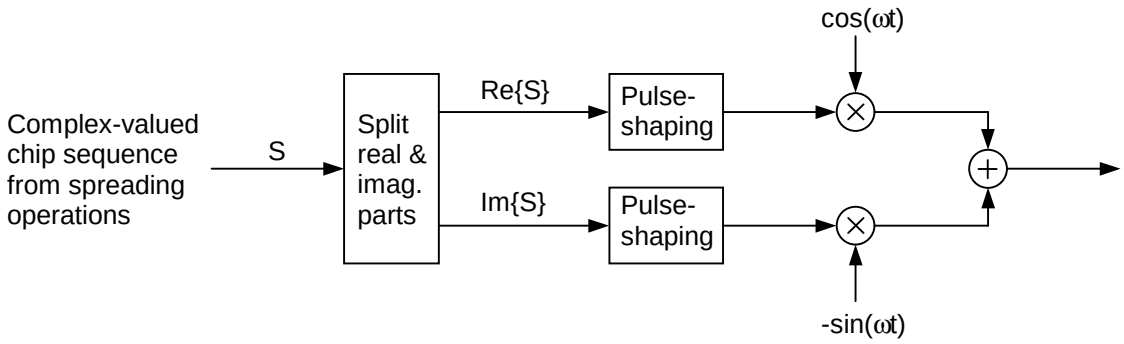

Figure 7.4: Uplink modulation and pulse shaping

Pulse shaping is performed through square root raised cosine filters with roll-off α=0,22 as shown in figure 7.4.

The impulse response of the pulse shaping is thus equal to:

$$p(t) = \frac{\sin\left(\pi \frac{t}{T_c}(1-\alpha)\right) + 4\alpha \frac{t}{T_c}\cos\left(\pi \frac{t}{T_c}(1+\alpha)\right)}{\pi \frac{t}{T_c}\left(1-\left(4\alpha \frac{t}{T_c}\right)^2\right)}$$

Where Tc is the chip period.

8 Radio transmission

8.1 Frequency bands and channel arrangement

The nominal channel spacing is either 5 MHz or 2,5 MHz or 325 kHz respectively for the 3,84 Mchip/s, 1,92 Mchip/s and 240 kchip/s.

The terminal shall be able to transmit on any frequency signalled in the SCT table in TS 102 721-6 [3]. The granularity for the frequency signalled in the SCT is 100 Hz.

8.2 Stability and accuracy requirements

For stability the terminal shall be locked either on a satellite Forward-Link (FL) carrier or on a Down-Link (DL) carrier from a terrestrial repeater.

It is assumed here that the terminal will generate the Tx carrier and clock frequencies based on a reference locked on the recovered FL (or DL) carrier clock. The relative accuracy with which the FL carrier clock frequency shall be recovered should be better than $3*10^{-7}$. This would allow the GW demodulator to reduce the search range for packet detection. At this regard it is assumed that the transmitted FL (or DL) carrier clock frequency is known with an accuracy of 10^{-8}. Also the GW knows and can pre-compensate the satellite Doppler and the frequency conversion error of the satellite transponders. It is thus expected that the major contributions to the frequency error seen by the GW demodulator are the reference clock accuracy of the terminal and the terminal caused Doppler.

The total frequency error at the GW side can be estimated as:

- Error in the FL reference clock recovered at the terminal: $3*10^{-7}$
- Uncompensated satellite Doppler: 10^{-8}
- Frequency accuracy of the transmitted FL carrier: 10^{-8}

- Gateway reference frequency accuracy: 10^{-8}
- Uncompensated Satellite Frequency Translation Error: $1,5*10^{-7}$
- Uncompensated Terminal Doppler: $2,4*10^{-7}$

NOTE: This assumes no Doppler compensation for vehicle speed up to 130 km/h. Further, Doppler from both the FL and RL is considered. Faster vehicles (i.e. trains) are assumed to have Doppler pre-compensation.

The total relative frequency error of return link carriers relative to GW demodulator reference is $7,2*10^{-7}$ which amounts to a maximum carrier frequency error of less than 1,5 kHz for the MSS S-band..

8.2.1 Frequency and symbol timing stability and accuracy

The modulated carrier frequency shall be accurate to within ±0,5 PPM observed over a period of one frame compared to the carrier frequency received from the satellite or a terrestrial repeater. The terminal shall use the same frequency source for both RF frequency generation and the chip clock.

8.2.2 Power stability and accuracy

The accuracy with which the output power is set shall be better than 2 dB.

The terminal shall be able to set its output power in a range of 30 dB with monotonic steps of 0,5 dB from [P_{max} -30 dB to P_{max}].

When not transmitting, the output power shall be below -56 dBm.

8.3 Transmitter characteristics

8.3.1 Power output characteristics and power class

NOTE: Typical EIRP values are defined in TS 102 721-1 [4].

8.3.2 Transmit polarization

8.3.2.1 Satellite Access

Circular polarization shall be supported by vehicular terminals. Hand-held terminals, if allowed by the system operator may use linear polarization but QoS may be reduced with respect to vehicular terminals.

8.3.2.2 Terrestrial Access

Linear polarization will be used by the CGC. This does not imply that vehicular terminals shall use an ad-hoc antenna for CGC (although this may improve the link budget as the optimal antenna patterns for satellite and terrestrial paths are different).

8.3.3 Unwanted emissions

Adjacent Channel Leakage power Ratio (ACLR) (defined as the ratio of the Square Root Raised Cosine (SRRC) filtered mean power centred on the assigned channel frequency to the SRRC filtered mean power centred on an adjacent channel frequency) shall be higher than 33 dB for the first adjacent channel slot and higher than 43 dB for the 2^{nd} adjacent channel slot. The channel slot widths are 5 MHz, 2,5 MHz and 325 kHz respectively for the 3,84 Mchip/s, 1,92 Mchip/s and 240 kchip/s options.

8.4 Transmit procedure

The following transmit procedure is foreseen.

(1) The terminal randomly selects a preamble spreading code from the set of available spreading codes (if more than one is available). The random function shall provide a uniform distribution of the selected codes.

(2) The terminal sets the transmit power in order to get an EIRP value depending on the value of parameter Rl_tx_EIRP_flag broadcasted through the SAT table in TS 102 721-6 [3] according to the following options:

- Rl_tx_EIRP = 00

  No terminal can transmit

- Rl_tx_EIRP = 01:

  The terminal shall set its transmitted EIRP to the value P computed as $P = L + N_{SAT} + K - R_{rand}$ dBm.

  where L is the path loss in dB (also including fading due to e.g. shadowing), estimated from the SNR measured on the FL carrier using the value of the forward link EIRP which is broadcasted through the SAT Table; N_{SAT} is the noise+interference power level at the satellite receiver input expressed in dBm, which is broadcasted through the SDT Table. K is a constant value broadcasted through the SAT Table.

NOTE: The algorithm parameters are system-dependent and depends on the type of signal processing implemented at the gateway side (which is not part of this multi-part deliverable).

  The path loss, L, actually also implicitly includes the geographic advantage of the terminal (i.e. the position of the terminal within the Tx antenna beam).

  The parameter K is equal to $C/(N_0+ I_0)|_T - G_S$ where $C/(N_0+ I_0)|_T$ is the target value used for the desired $C/(N_0+ I_0)$ at the satellite transponder input. G_S is the satellite antenna gain at EOC.

  R_{rand} is a random value between 0 and R_{max} which is extracted by the terminal according to a uniform distribution. R_{max} is also designated on the SAT Table broadcasted on the FL with the purpose of optimising the demodulator performance. If the value of P exceeds the terminal capability no transmission shall take place as this is an indication of excessive channel fading. A CGC could be used in such a case if available.

- Rl_tx_EIRP = 10

  The terminal transmits with maximum EIRP if the measured SNR from the FL carrier is greater than $SNR_{LOS} - R_{max}$, being R_{max} a parameter broadcasted through the SAT table and

$$SNR_{LOS} = \max\left\{SNR_{ref}; \frac{1}{N}\sum_{i=1}^{N} SNR_N\right\}$$

  where SNR_{ref} is a pre-configured minimum system guaranteed LOS received signal power value and SNR_N are the last N best SNR estimates. $N = 10$ is recommended value.

- Rl_tx_EIRP = 11

  The terminal transmit always with maximum EIRP

(3) Once a packet has been transmitted, the transmission time of next packet depends whether an acknowledgment is required or not and, if required, if cumulative acknowledgments are allowed.

If no acknowledgment is required, the next packet transmission may not be done before a time interval computed by using the given persistence value and back-off time which are signalled for the terminal user group and for the given congestion status on the SAT table.

If an ACK per packet is required no further packet transmission may happen before the reception of the ACK.

If no ACK is received before the associated message Time_Out, a random draw using the current dynamic persistence value and the actual back-off time is done for computing the packet retransmission time. If the retransmitted packet is not acknowledged and the number of allowed retransmission is less than the allowed retransmission number, the dynamic persistence value is halved.

8.5 Tx/Rx states

The terminal transmit/receive states are as follows:

- Idle (terminal disabled or switched off).
- Search (transmission disabled).
- Satellite Rx synchronized status (transmission enabled upon reception of the SCT, SAT and SDT signalling tables, see [3]).
- CGC Rx Synchronized status (transmission enabled upon reception of the SCT, SAT and SDT signalling tables, see [3]).

An additional state may be considered for terminals able to monitor both satellite and repeater carriers at the same time. Figure 8.1 shows the state transitions (transitions to the Idle state are not shown for simplicity).

The Search state is the initial state of the terminal after switch-on. In that state a search of FL satellite or terrestrial carriers is performed. Priority is to search for satellite first, then revert to terrestrial if no satellite carrier is found. The search state is re-entered after loss of Rx Synchronization (i.e. carrier outage) lasting for more than T_{Sat} or T_{CGC} seconds. Suggested values for T_{Sat} and T_{CGC} are respectively 3 s and 1 s.

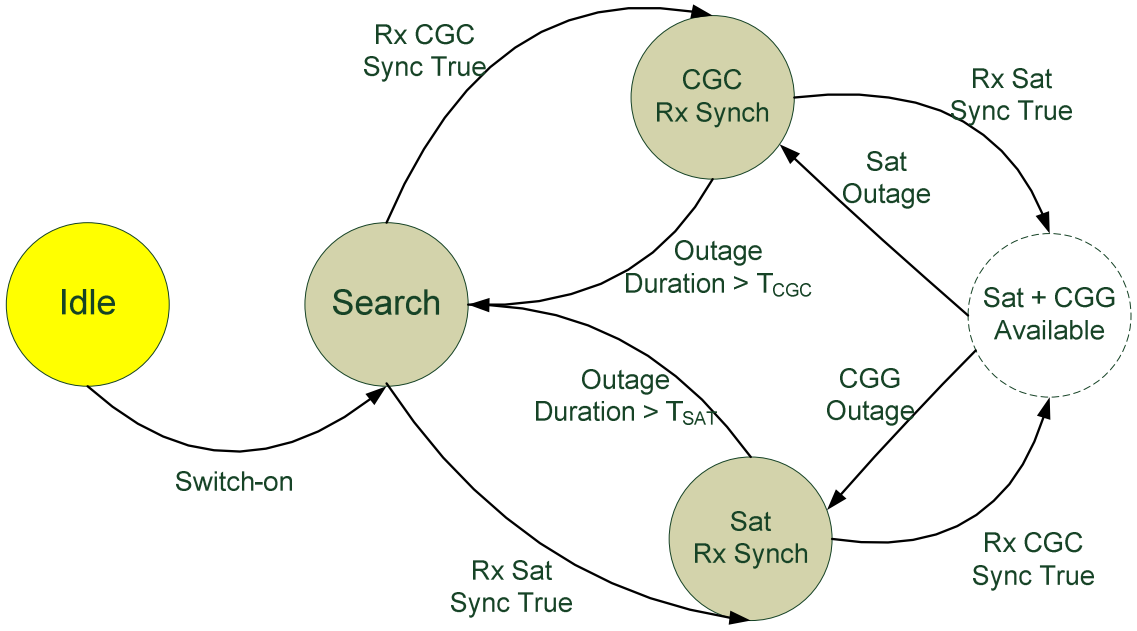

Figure 8.1: Tx/Rx states

9 Physical layer measurements

Radio characteristics are measured and reported to higher layers and the network. Such measurements are for further study.

Annex A (informative): Generation of Complex Complementary Golay Codes

The pair of complementary sequences can be generated by the following Haskell code:

```
golayAux n w p a b
|(n < 0) = ([],[])
|(n == 0) = (a,b)
|otherwise =  golayAux (n-1) w p d e
    where
        i = n-1
        c = map (\x -> x*(w!!i)) (rightCyclicShift (p!!i) b)
        d = zipWith (+) a c
        e = zipWith (-) a c

golay n w p =  golayAux n  w p       ([1] ++ [0 | i<- [1 .. (2^n)-1]])
                                     ([1] ++ [0 | i<- [1 .. (2^n)-1]])
```

The function golay uses a helper function golayAux for doing the actual computation. Functions used in golayAux are Haskell standard library functions (map and zipWith) or function whose purpose is self evident (i.e. rightCyclicShift).

The sequences in the specifications were generated by calling the golay function with the following arguments (and successive 45° rotation):

N=16

golay 4 [0:+1 , 0:+1, 0:+1, 0:+1] [2, 1, 8, 4]

N=128

golay 7 [0:+1 , -1, 0:+1, -1, 0:+1, -1, -1] [16; 1; 64; 4; 8; 2; 32]

N=256

golay 8 [0:+1,0:+1,0:+1,0:+1, -1, 0:+1,0:+1, -1] [16,8,4,32,2,64,1,128]

History

Document history		
V1.1.1	December 2011	Publication