CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.213	CR	**024r1**	Current Version:	3.1.1

GSM (AA.BB) or 3G (AA.BBB) specification number ↑   *↑ CR number as allocated by MCC support team*

For submission to:	RAN#7	for approval	**X**	strategic		*(for SMG*
list expected approval meeting # here ↑		for information		non-strategic		*use only)*

Form: CR cover sheet, version 2 for 3GPP and SMG   The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc

Proposed change affects: (U)SIM [ ]   ME **X**   UTRAN / Radio **X**   Core Network [ ]
(at least one should be marked with an X)

Source:	Siemens		**Date:**	2000-2-21

Subject:	Editorial changes to 25.213

Work item:	

Category: F   Correction   **X**   **Release:**   Phase 2   [ ]
              A   Corresponds to a correction in an earlier release   [ ]   Release 96   [ ]
(only one category   B   Addition of feature   [ ]   Release 97   [ ]
shall be marked   C   Functional modification of feature   [ ]   Release 98   [ ]
with an X)   D   Editorial modification   [ ]   Release 99   **X**
                                                                                   Release 00   [ ]

Reason for change:	correction from WG1 feedback

Clauses affected:	4.2.1, 4.2.2.1, 4.2.3.1

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:	
Other GSM core specifications	[ ]	→ List of CRs:	
MS test specifications	[ ]	→ List of CRs:	
BSS test specifications	[ ]	→ List of CRs:	
O&M specifications	[ ]	→ List of CRs:	

Other comments:	

help.doc

<--------- double-click here for help and instructions on how to create a CR.

4.2 Spreading

4.2.1 DPCCH/DPDCH

Figure 1 illustrates the principle of the uplink spreading of DPCCH and DPDCHs. The binary DPCCH and DPDCHs to be spread are represented by real-valued sequences, i.e. the binary value "0" is mapped to the real value +1, while the binary value "1" is mapped to the real value –1. The DPCCH is spread to the chip rate by the channelization code c_c, while the n:th DPDCH called $DPDCH_n$ is spread to the chip rate by the channelization code $c_{d,n}$. One DPCCH and up to six parallel DPDCHs can be transmitted simultaneously, i.e. $\theta\underline{1} \le n \le 6$.
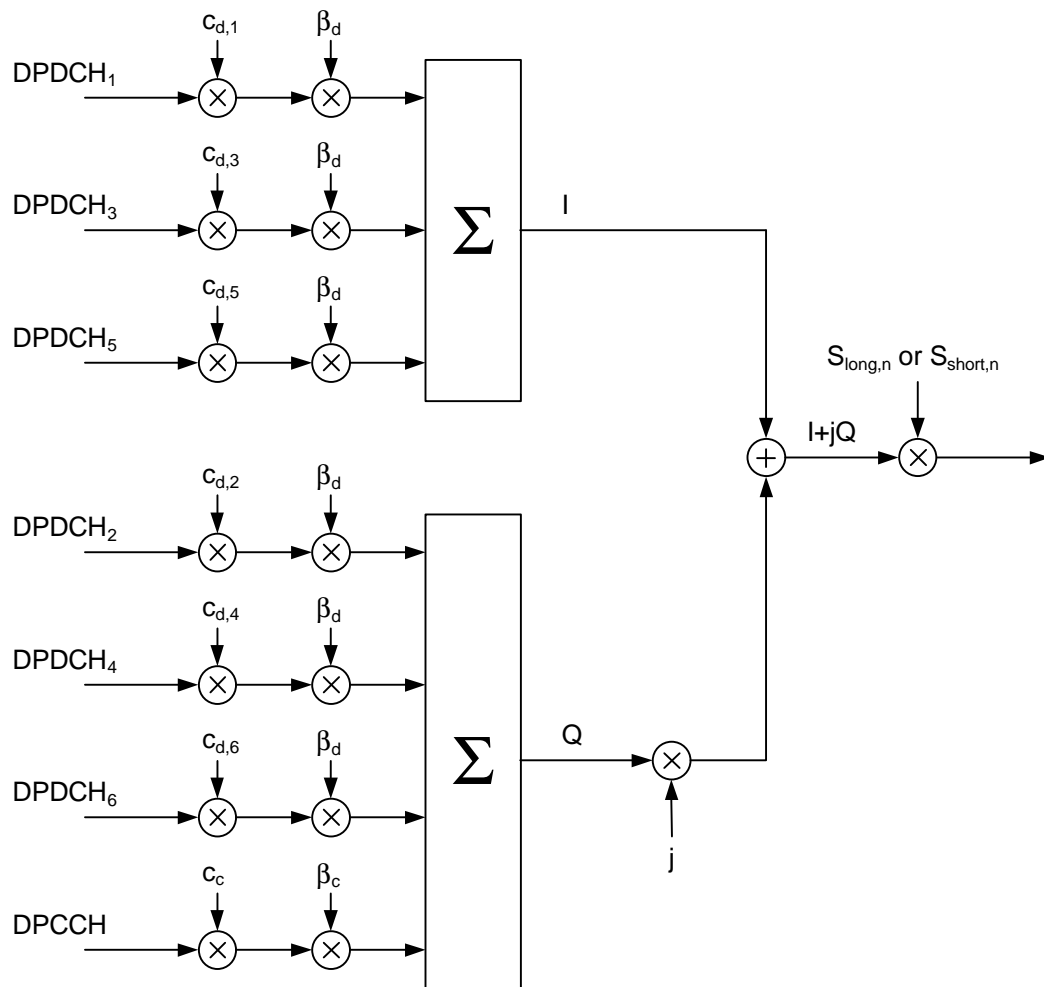

Figure 1: Spreading for uplink DPCCH and DPDCHs

After channelization, the real-valued spread signals are weighted by gain factors, β_c for DPCCH and β_d for all DPDCHs.

At every instant in time, at least one of the values β_c and β_d has the amplitude 1.0. The β-values are quantized into 4 bit words. The quantization steps are given in table 1.

Table 1: The quantization of the gain parameters

Signalling values for β_c and β_d	Quantized amplitude ratios β_c and β_d
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

After the weighting, the stream of real-valued chips on the I- and Q-branches are then summed and treated as a complex-valued stream of chips. This complex-valued signal is then scrambled by the complex-valued scrambling code $S_{long,n}$ or $S_{short,n}$, depending on if long or short scrambling codes are used. The scrambling code is applied aligned with the radio frames, i.e. the first scrambling chip corresponds to the beginning of a radio frame.

4.2.2   PRACH

4.2.2.1   PRACH preamble part

The PRACH preamble part consists of a complex-valued code, described in section 4.3.3.

4.2.2.2   PRACH message part

Figure 2 illustrates the principle of the spreading and scrambling of the PRACH message part, consisting of data and control parts. The binary control and data parts to be spread are represented by real-valued sequences, i.e. the binary value "0" is mapped to the real value +1, while the binary value "1" is mapped to the real value –1. The control part is spread to the chip rate by the channelization code c_c, while the data part is spread to the chip rate by the channelization code c_d.
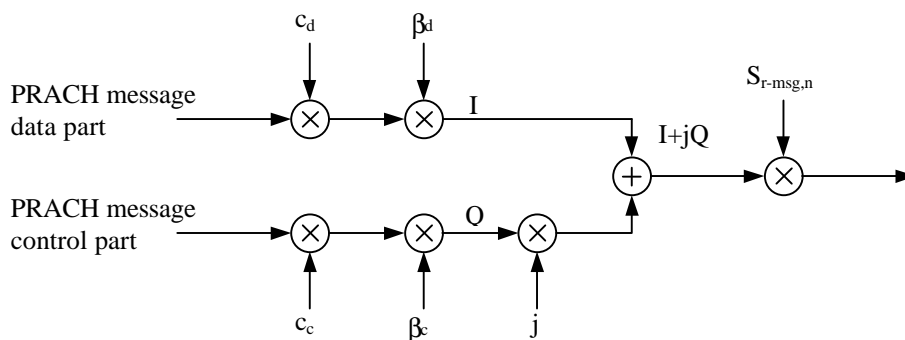

Figure 2: Spreading of PRACH message part

After channelization, the real-valued spread signals are weighted by gain factors, β_c for the control part and β_d for the data part. At every instant in time, at least one of the values β_c and β_d has the amplitude 1.0. The β-values are quantized into 4 bit words. The quantization steps are given in section 4.2.1.

After the weighting, the stream of real-valued chips on the I- and Q-branches are treated as a complex-valued stream of chips. This complex-valued signal is then scrambled by the complex-valued scrambling code $S_{r-msg,n}$. The 10 ms scrambling code is applied aligned with the 10 ms message part radio frames, i.e. the first scrambling chip corresponds to the beginning of a message part radio frame.

4.2.3   PCPCH

4.2.3.1   PCPCH preamble part

The PCPCH preamble part consists of a complex-valued code, described in section 4.3.4.

4.3.1.3 Code allocation for PRACH message part

The preamble signature *s*, $0\underline{1} \leq s \leq 1\underline{5}6$, points to one of the 16 nodes in the code-tree that corresponds to channelization codes of length 16. The sub-tree below the specified node is used for spreading of the message part. The control part is spread with the channelization code c_c (as shown in section 4.2.2.2) of spreading factor 256 in the lowest branch of the sub-tree, i.e. $c_c = C_{ch,256,m}$ where m = $16\underline{\times}(s\ \text{1}) + 15$. The data part uses any of the channelization codes from spreading factor 32 to 256 in the upper-most branch of the sub-tree. To be exact, the data part is spread by channelization code $c_d = C_{ch,SF,m}$ and SF is the spreading factor used for the data part and m = $SF\underline{\times}(s\ \text{1})/16$.

4.3.2.3 Short scrambling sequence

The short scrambling sequences $c_{short,1,n}(i)$ and $c_{short,2,n}(i)$ are defined from a sequence from the family of periodically extended S(2) codes.

Let $n_{23}n_{22}\ldots n_0$ be the 24 bit binary representation of the code number n.

The n:th quaternary S(2) sequence $z_n(i)$, $0 \leq n \leq 16777215$, is obtained by modulo 4 addition of three sequences, a quaternary sequence $a(i)$ and two binary sequences $b(i)$ and $d(i)$, where the initial loading of the three sequences is determined from the code number n. The sequence $z_n(i)$ of length 255 is generated according to the following relation:

$z_n(i) = a(i) + 2b(i) + 2d(i)$ modulo 4, $i = 0, 1, \ldots, 254$,

where the quaternary sequence $a(i)$ is generated recursively by the polynomial $g_0(x)= x^8+x^5+3x^3+x^2+2x+1$ as

$a(0) = 2n_0 + 1$ modulo 4,

$a(i) = 2n_i$ modulo 4, $i = 1, 2, \ldots, 7$,

$a(i) = 3a(i\text{-}3) + a(i\text{-}5) + 3a(i\text{-}6) + 2a(i\text{-}7) + 3a(i\text{-}8)$ modulo 4, $i = 8, 9, \ldots, 254$,

and the binary sequence $b(i)$ is generated recursively by the polynomial $g_1(x)= x^8+x^7+x^5+x+1$ as

$b(i) = n_{8+i}$ modulo 2, $i = 0, 1, \ldots, 7$,

$b(i) = b(i\text{-}1) + b(i\text{-}3) + b(i\text{-}7) + b(i\text{-}8)$ modulo 2, $i = 8, 9, \ldots, 254$,

and the binary sequence $\underline{d}(i)$ is generated recursively by the polynomial $g_2(x)= x^8+x^7+x^5+x^4+1$ as

$d(i) = n_{16+i}$ modulo 2, $i = 0, 1, \ldots, 7$,

$d(i) = d(i\text{-}1) + d(i\text{-}3) + d(i\text{-}4) + d(i\text{-}8)$ modulo 2, $i = 8, 9, \ldots, 254$.

The sequence $z_n(i)$ is extended to length 256 chips by setting $z_n(255) = z_n(0)$.

The mapping from $z_n(i)$ to the real-valued binary sequences $c_{short,1,n}(i)$ and $c_{short,2,n}(i)$, , $i = 0, 1, \ldots, 255$ is defined in Table 2.

Table 2. Mapping from $z_n(i)$ to $c_{short,1,n}(i)$ and $c_{short,2,n}(i)$, $i = 0, 1, \ldots, 255$.

$z_n(i)$	$c_{short,1,n}(i)$	$c_{short,2,n}(i)$
0	+1	+1
1	-1	+1
2	-1	-1
3	+1	-1

Finally, the complex-valued short scrambling sequence $C_{short, n}$, is defined as

$$C_{short,n}(i) = c_{short,1,n}(i \bmod 256)\left(1 + j(-1)^i c_{short,2,n}\left(2\lfloor (i \bmod 256)/2 \rfloor\right)\right)$$

where $i = 0, 1, 2, \ldots$ and $\lfloor \rfloor$ denotes rounding to nearest lower integer.

An implementation of the short scrambling sequence generator for the 255 chip sequence to be extended by one chip is shown in Figure 6.
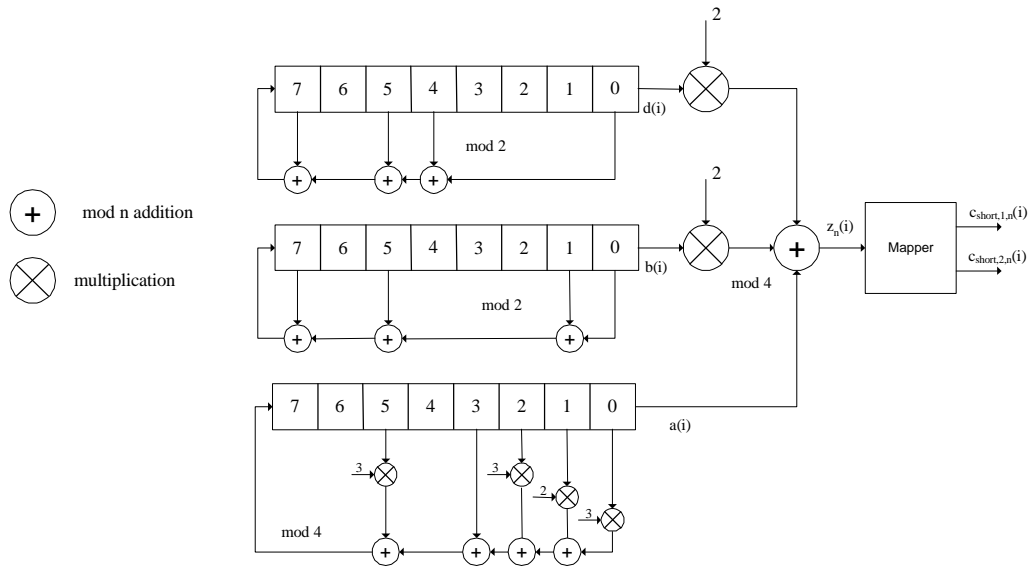
Figure 6. Uplink short scrambling sequence generator for 255 chip sequence.

4.3.2.6 PCPCH message part scrambling code

The set of scrambling codes used for the PCPCH message part are 10 ms long, cell-specific and have a one-to-one correspondence to the signature sequences and the access sub-channels used by the access preamble part. Both long or short scrambling codes can be used to scramble the CPCH message part.

The n:th PCPCH message part scrambling code, denoted $S_{c\text{-msg},n}$, is based on the scrambling sequence and is defined as

In the case when the long scrambling codes are used,

$$S_{c\text{-msg},n}(i) = C_{long,n}(i + 8192), \ i = 0, 1, \ldots, 38399$$

where the lowest index corresponds to the chip transmitted first in time and $C_{long,n}$ is defined in section 4.3.2.2.

In the case when the access resources are shared between the RACH and CPCH, then $S_{c\text{-msg},n}$ is defined as

$$S_{c\text{-msg},n}(i) = C_{long,n}(i + 4096), \ i = 0, 1, \ldots, 38399$$

where the lowest index corresponds to the chip transmitted first in time and $C_{long,n}$ is defined in section 4.3.2.2.

In the case the short scrambling codes are used,

$$S_{c\text{-msg},n}(i) = C_{short,n}(i), \ i = 0, 1, \ldots, 38399$$