3GPP/TSG RAN WG1 Meeting #12
Seoul, Korea, April 10th-14th 2000

Document R1-00-0489
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

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

25.222 CR 033 Current Version: 3.2.0

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

For submission to: TSG RAN #8
list expected approval meeting # here ↑

for approval	X	strategic	
for information		non-strategic	

(for SMG 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:
(at least one should be marked with an X)

(U)SIM	ME	UTRAN / Radio	Core Network
	X	X	

Source: Nokia **Date:** 2000-04-04

Subject: Editorial changes for 25.222

Work item: UTRAN

Category:
(only one category shall be marked with an X)

F	Correction	
A	Corresponds to a correction in an earlier release	
B	Addition of feature	
C	Functional modification of feature	
D	Editorial modification	X

Release:

Phase 2	
Release 96	
Release 97	
Release 98	
Release 99	
Release 00	X

Reason for change: Clarification for definition of q_i and for pruning in section 4.2.3.2.3

Clauses affected: 4.2.3.2.3, 4.2.3.2.3.2 and 4.2.3.2.3.3

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.

4.2.3.2.3 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 $x_1, x_2, x_3, \ldots, x_K, x_{K+1}, \ldots, x_{RC}$, where K is the integer number of the bits and takes one value of $40 \leq K \leq 5114$. The relation between the bits input to the Turbo code internal interleaver and the bits input to the channel coding is defined by $x_k = o_{irk}$ for k=1,2,…,K and $x_k = 0$ for k=K+1, K+2,…,RC , and $K = K_i$.

The following section specific symbols are used in sections 4.2.3.2.3.1 – 4.2.3.4.3.3:

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
q_j	Minimum prime integers
r_j	Permuted prime integers
$T(j)$	Inter-row permutation pattern
$U_j(i)$	Intra-row permutation pattern
i	Index of matrix
j	Index of matrix
k	Index of bit sequence

4.2.3.2.3.1 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

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

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 ($481 \leq K \leq 530$) then

$p = 53$ and $C = p$.

else

Find minimum prime p such that

$(p + 1) - K/R \geq 0$,

and determine C such that

if (p - $K/R \geq 0$) then

if (p - 1 - $K/R \geq 0$) 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:

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

4.2.3.2.3.2 Intra-row and inter-row permutations

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

(1) Select a primitive root v from table 2.

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

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

(3) Selection of the consecutive minimum prime integers: assign ~~Let~~ q_0 = 1 be the first term prime ~~integer in {q_j},~~ and then select ~~the consecutive minimum prime integers {q_j}~~ to be a least prime number ~~(j = 1, 2, …, R - 1)~~ such that

g.c.d{q_j, p - 1} = 1, $q_j > 6$, and $q_j > q_{(j-1)}$, for each j = 1, 2, …, R – 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, …, R - 1,

where $T(j)$ indicates the original row position of the j-th permuted row, and $T(j)$ 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.

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

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, …, 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, …, (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, …, (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, …, (p - 2),

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

end if

Table 2: 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		

4.2.3.2.3.3 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, o_{irk} to the channel coding, 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

$$R \times C - K.$$