TSG-RAN Meeting #14 ***RP-010747***
Kyoto, Japan, 11 – 14, December, 2001

Title: **Agreed CR (Rel-4) to TS 25.222**

Source: **TSG-RAN WG1**

Agenda item: **8.1.4**

No.	Spec	CR	Rev	R1 T-doc	Subject	Release	Cat	W/I Code	V_old	V_new
4	25.222	059	-	R1-01-0807	Bit scrambling for TDD	Rel-4	F	LCRTDD-Phys	4.1.0	4.2.0
5	25.222	061	-	R1-01-1149	Corrections in clause 4.1 and 4.2 of TS 25.222	Rel-4	F	LCRTDD-Phys	4.1.0	4.2.0

3GPP TSG RAN Meeting #14
Kyoto, Japan, 11th-14th, December, 2001

R1-01-0807

CR-Form-v3

CHANGE REQUEST

⌘	**25.222**	**CR**	**059**	⌘ rev	**-**	⌘ Current version:	**4.1.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM [ ] ME/UE [X] Radio Access Network [X] Core Network [ ]

Title:	⌘	Bit Scrambling for TDD		
Source:	⌘	TSG RAN WG1		
Work item code:	⌘	LCRTDD-Phys	***Date:*** ⌘	01.11.01
Category:	⌘	**F**	***Release:*** ⌘	REL-4
		Use one of the following categories: ***F** (essential correction)* ***A** (corresponds to a correction in an earlier release)* ***B** (Addition of feature),* ***C** (Functional modification of feature)* ***D** (Editorial modification)* Detailed explanations of the above categories can be found in 3GPP TR 21.900.	*Use one of the following releases:* *2 (GSM Phase 2)* *R96 (Release 1996)* *R97 (Release 1997)* *R98 (Release 1998)* *R99 (Release 1999)* *REL-4 (Release 4)* *REL-5 (Release 5)*	

Reason for change:	⌘	In specific situations, when the transmitted data contain a lot of the same symbols the data bursts will contain a DC offset that leads to not acceptable degradations in the link level performance if it is discarded.
Summary of change:	⌘	Bit Scrambling is used to avoid possible DC offsets. The section on the coding of the PICH for 1.28 Mcps TDD has been moved from TS 25.221 to TSS 25.222 according to the respective 3.84 Mcps TDD sections.
Consequences if not approved:	⌘	Not tolerable restrictions in implementation or not acceptable degradations in the link level performance. Inconsistency in set of specification, since Bit Scrambling introduced for 3.84 Mcps TDD and common sections have been modified and do not longer "fit" to 1.28 Mcps TDD.

Clauses affected:	⌘	4.2; new section 4.4.3, section numbering changed from 4.4.3 to 4.4.4		
Other specs affected:	⌘	[X] Other core specifications	⌘	TS25.221 CR 059
		[ ] Test specifications		
		[ ] O&M Specifications		
Other comments:	⌘			

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

4.2 Transport channel coding/multiplexing

Figure 1 illustrates the overall concept of transport-channel coding and multiplexing. Data arrives to the coding/multiplexing unit in form of transport block sets, once every transmission time interval. The transmission time interval is transport-channel specific from the set {5 ms$^{(*1)}$, 10 ms, 20 ms, 40 ms, 80 ms}.

Note: $^{(*1)}$ may be applied for PRACH for 1.28 Mcps TDD

The following coding/multiplexing steps can be identified:

- add CRC to each transport block (see subclause 4.2.1);
- TrBk concatenation / Code block segmentation (see subclause 4.2.2);
- channel coding (see subclause 4.2.3) ;
- radio frame size equalization (see subclause 4.2.4);
- interleaving (two steps, see subclauses 4.2.5 and 4.2.10);
- radio frame segmentation (see subclause 4.2.6);
- rate matching (see subclause 4.2.7);
- multiplexing of transport channels (see subclause 4.2.8);
- bit scrambling (see subclause 4.2.9);
- physical channel segmentation (see subclause 4.2.10);
- sub-frame segmentation(see subclause 4.2.12 only for 1.28Mcps TDD)
- mapping to physical channels (see subclause 4.2.13).

The coding/multiplexing steps for uplink and downlink are shown in figures 1 and 2.

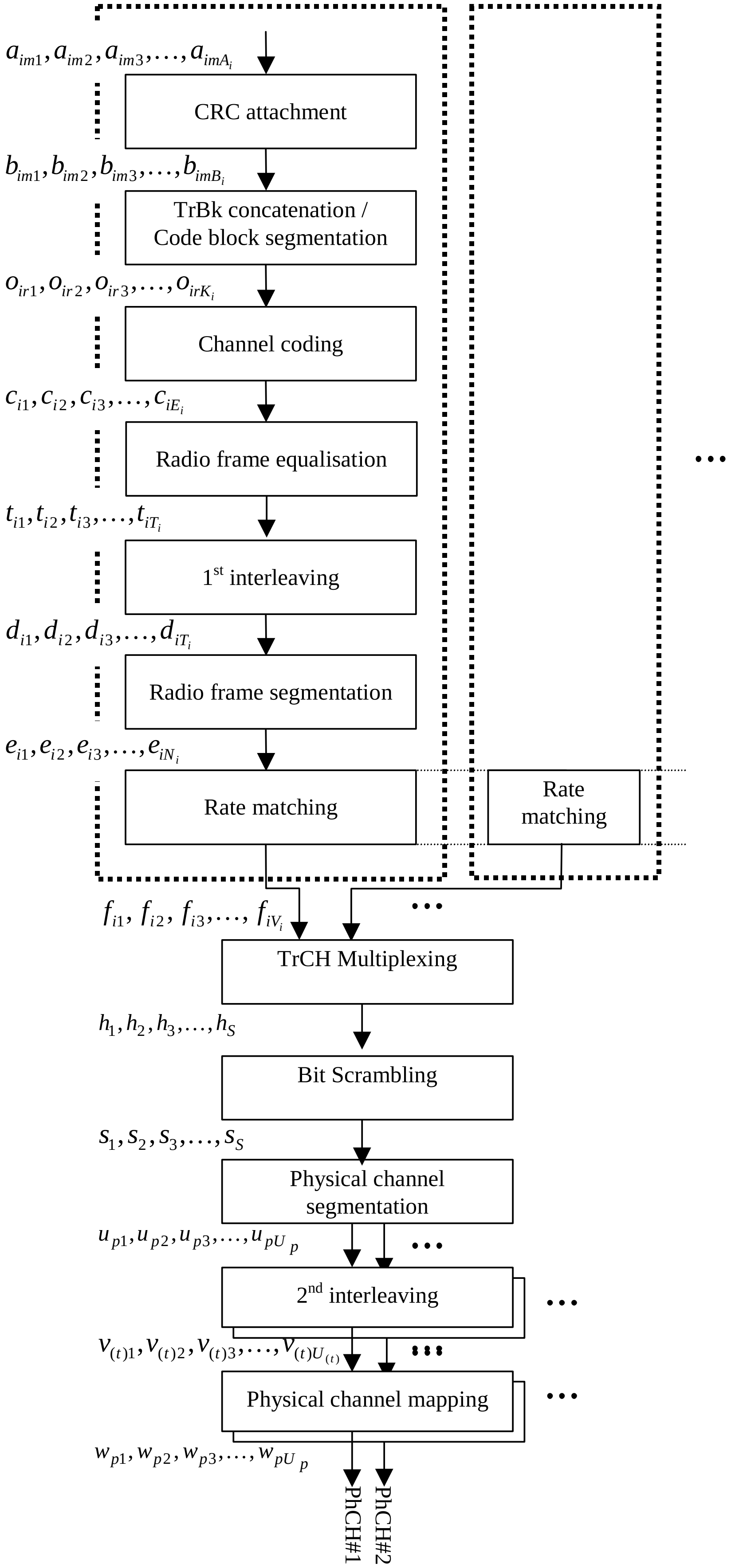
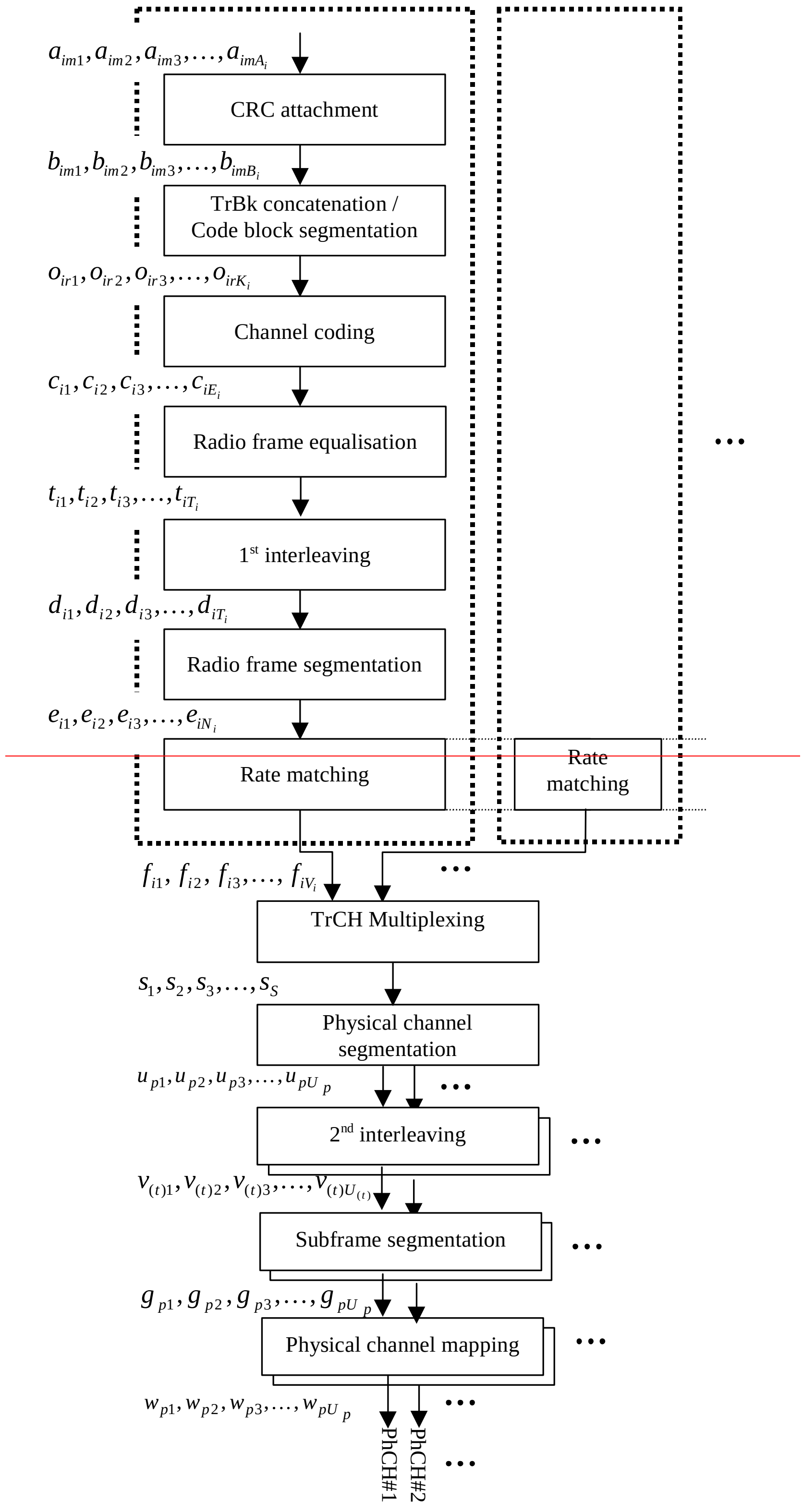


Figure 1: Transport channel multiplexing structure for uplink and downlink for 3.84Mcps TDD



$a_{im1}, a_{im2}, a_{im3}, \ldots, a_{imA_i}$
CRC attachment
$b_{im1}, b_{im2}, b_{im3}, \ldots, b_{imB_i}$
TrBk concatenation / Code block segmentation
$o_{ir1}, o_{ir2}, o_{ir3}, \ldots, o_{irK_i}$
Channel coding
$c_{i1}, c_{i2}, c_{i3}, \ldots, c_{iE_i}$
Radio frame equalisation
$t_{i1}, t_{i2}, t_{i3}, \ldots, t_{iT_i}$
1st interleaving
$d_{i1}, d_{i2}, d_{i3}, \ldots, d_{iT_i}$
Radio frame segmentation
$e_{i1}, e_{i2}, e_{i3}, \ldots, e_{iN_i}$
Rate matching
Rate matching
$f_{i1}, f_{i2}, f_{i3}, \ldots, f_{iV_i}$
TrCH Multiplexing
$s_1, s_2, s_3, \ldots, s_S$
Physical channel segmentation
$u_{p1}, u_{p2}, u_{p3}, \ldots, u_{pU_p}$
2nd interleaving
$v_{(t)1}, v_{(t)2}, v_{(t)3}, \ldots, v_{(t)U_{(t)}}$
Subframe segmentation
$g_{p1}, g_{p2}, g_{p3}, \ldots, g_{pU_p}$
Physical channel mapping
$w_{p1}, w_{p2}, w_{p3}, \ldots, w_{pU_p}$
PhCH#1
PhCH#2

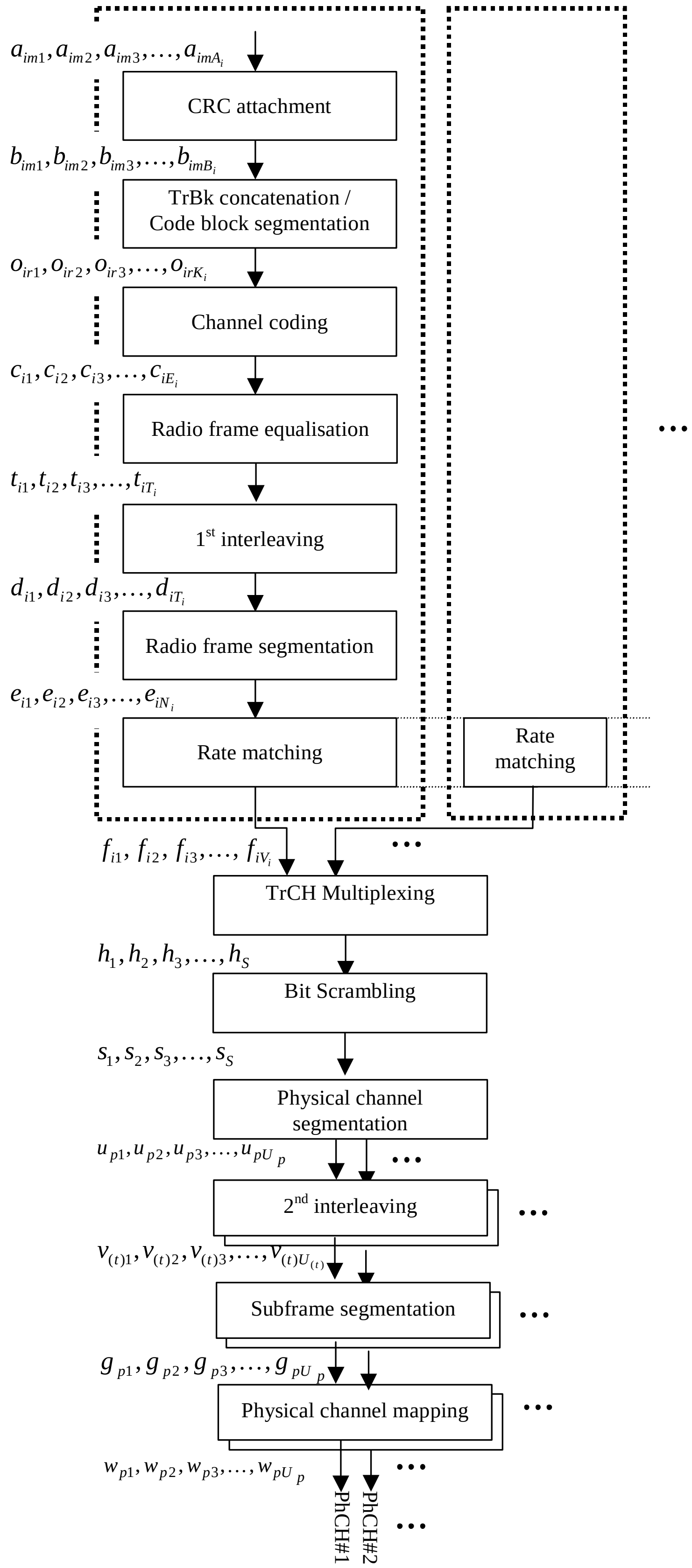


Figure 2: Transport channel multiplexing structure for uplink and downlink of 1.28Mcps TDD

Primarily, transport channels are multiplexed as described above, i.e. into one data stream mapped on one or several physical channels. However, an alternative way of multiplexing services is to use multiple CCTrCHs (Coded Composite Transport Channels), which corresponds to having several parallel multiplexing chains as in figures 1 and 2, resulting in several data streams, each mapped to one or several physical channels.

4.4.3 Coding and Bit Scrambling of the Paging Indicator

The paging indicator P_q, $q = 0, ..., N_{PI}-1$, $P_q \in \{0, 1\}$ is an identifier to instruct the UE whether there is a paging message for the groups of mobiles that are associated to the PI, calculated by higher layers, and the associated paging indicator P_q. The length L_{PI} of the paging indicator is $L_{PI}=2$, $L_{PI}=4$ or $L_{PI}=8$ symbols. $N_{PIB} = 2*N_{PI}*L_{PI}$ bits are used for the paging indicator transmission in one radio frame. The mapping of the paging indicators to the bits e_i, $i = 1, ..., N_{PIB}$ is shown in table 19.

Table 19: Mapping of the paging indicator

P_q	Bits $\{e_{2L_{PI}*q+1}, e_{2L_{PI}*q+2}, ..., e_{2L_{PI}*(q+1)}\}$	Meaning
0	{0, 0, ..., 0}	There is no necessity to receive the PCH
1	{1, 1, ..., 1}	There is the necessity to receive the PCH

If the number S of bits in one radio frame available for the PICH is bigger than the number N_{PIB} of bits used for the transmission of paging indicators, the sequence $e = \{e_1, e_2, ..., e_{NPIB}\}$ is extended by $S-N_{PIB}$ bits that are set to zero, resulting in a sequence $h = \{h_1, h_2, ..., h_S\}$:

$$h_k = e_k, \quad k = 1, ..., N_{PIB}$$
$$h_k = 0, \quad k = N_{PIB}, ..., S$$

The bits h_k, $k = 1, ..., S$ on the PICH then undergo bit scrambling as defined in section 4.2.9.

The bits s_k, $k = 1, ..., S$ output from the bit scrambler are then transmitted over the air as shown in [7].

4.4.~~3~~4 Coding of the Fast Physical Access Channel (FPACH) information bits

The FPACH burst is composed by 32 information bits which are block coded and convolutional coded, and then delivered in one sub-frame as follows:

1. The 32 information bits are protected by 8 parity bits for error detection as described in sub-clause 4.2.1.1.

2. Convolutional code with constraint length 9 and coding rate ½ is applied as described in sub-clause 4.2.3.1. The size of data block c(k) after convolutional encoder is 96 bits.

3. To adjust the size of the data block c(k) to the size of the FPACH burst, 8 bits are punctured as described in sub-clause 4.2.7 with the following clarifications:

   - $N_{i;j}$ =96 is the number of bits in a radio sub-frame before rate matching

   - $\Delta N_{i,,j}$ = -8 is the number of bits to punctured in a radio sub-frame

   - $e_{ini} = a \times N_{ij}$

   The 88 bits after rate matching are then delivered to the intra-frame interleaving.

4. The bits in input to the interleaving unit are denoted as {x(0), …, x(87)}. The coded bits are block rectangular interleaved according to the following rule: the input is written row by row, the output is read column by column.

$$\begin{bmatrix} x(0) & x(1) & x(2) & \ldots\ x(7) \\ x(8) & x(9) & x(10) & \ldots x(15) \\ \vdots & \vdots & \vdots & \ldots\ \vdots \\ x(80) & x(81) & x(82) & \ldots x(87) \end{bmatrix}$$

   Hence, the interleaved sequence is denoted by y (i) and are given by:

y(0), y(1), …, y(87)=x(0), x(8), …,x(80),x(1), …, x(87).

3GPP TSG RAN Meeting #14
Kyoto, Japan, 11th-14th, December, 2001

R1-01-1149

CR-Form-v4

CHANGE REQUEST

⌘ **25.222**	**CR 061**	⌘ rev **-**	⌘	Current version: **4.1.0**	⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM [ ] ME/UE [X] Radio Access Network [X] Core Network [ ]

Title: ⌘	Corrections in clause 4.1 and 4.2 of TS 25.222
Source: ⌘	TSG RAN WG1
Work item code: ⌘	LCRTDD-Phys
Date: ⌘	Nov.12th, 2001
Category: ⌘	**F**
Release: ⌘	REL-4

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
REL-4 (Release 4)
REL-5 (Release 5)

Reason for change: ⌘	The description about the maximum number of timeslot for UTRA-TDD is incomplete. The reference number of Table 7 in subclause 4.2.11.2 is incorrect.
Summary of change: ⌘	The exact number of maximum number of timeslots is deleted in order to avoid confusion. The reference number of Table 7 in subclause 4.2.11.2 is corrected as Table 8.
Consequences if not approved: ⌘	Confusion will rise because of the incomplete description of UTRA-TDD.

Clauses affected: ⌘	4.1 and 4.2.11.2
Other specs affected: ⌘	[ ] Other core specifications ⌘ [ ] Test specifications [ ] O&M Specifications
Other comments: ⌘	

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at: http://www.3gpp.org/3G_Specs/CRs.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://www.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2000-09 contains the specifications resulting from the September 2000 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

4.1 General

Data stream from/to MAC and higher layers (Transport block / Transport block set) is encoded/decoded to offer transport services over the radio transmission link. Channel coding scheme is a combination of error detection, error correcting (including rate matching), and interleaving and transport channels mapping onto/splitting from physical channels.

In the UTRA-TDD mode, the total number of basic physical channels (a certain time slot one spreading code on a certain carrier frequency) per frame is given by the maximum number of time slots ~~which is 15~~ and the maximum number of CDMA codes per time slot.

4.2.11.2 Timeslot related 2nd interleaving

In case of timeslot related 2nd interleaving, the bits input to the block interleaver are denoted by $x_{t,1}, x_{t,2}, x_{t,3}, \ldots, x_{t,U_t}$, where t refers to a certain timeslot, and U_t is the number of bits transmitted in this timeslot during the respective radio frame.

In each timeslot t the relation between $x_{t,k}$ and $u_{t,p,k}$ is given below with P_t refering to the number of physical channels within the respective timeslot:

$$x_{t,k} = u_{t,1,k} \quad k = 1, 2, \ldots, U_{t1}$$

$$x_{t,(k+U_{t1})} = u_{t,2,k} \qquad k = 1, 2, \ldots, U_{t2}$$

…

$$x_{t,(k+U_{t1}+\ldots+U_{t(P_t-1)})} = u_{t,P_t,k} \qquad k = 1, 2, \ldots, U_{tP_t}$$

The following steps have to be performed for each timeslot t, on which the respective CCTrCH is mapped:

(1) Assign C2 = 30 to be the number of columns of the matrix. The columns of the matrix are numbered 0, 1, 2, …, C2 - 1 from left to right.

(2) Determine the number of rows of the matrix, R2, by finding minimum integer R2 such that:

$U_t \leq \mathrm{R2} \times \mathrm{C2}$.

The rows of rectangular matrix are numbered 0, 1, 2, …, R2 - 1 from top to bottom.

(3) Write the input bit sequence $x_{t,1}, x_{t,2}, x_{t,3}, \ldots, x_{t,U_t}$ into the R2 × C2 matrix row by row starting with bit $y_{t,1}$ in column 0 of row 0:

$$\begin{bmatrix} y_{t,1} & y_{t,2} & y_{t,3} & \cdots & y_{t,\mathrm{C2}} \\ y_{t,(\mathrm{C2}+1)} & y_{t,(\mathrm{C2}+2)} & y_{t,(\mathrm{C2}+3)} & \cdots & y_{t,(2\times\mathrm{C2})} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ y_{t,((\mathrm{R2}-1)\times\mathrm{C2}+1)} & y_{t,((\mathrm{R2}-1)\times\mathrm{C2}+2)} & y_{t,((\mathrm{R2}-1)\times\mathrm{C2}+3)} & \cdots & y_{t,(\mathrm{R2}\times\mathrm{C2})} \end{bmatrix}$$

where $y_{t,k} = x_{t,k}$ for k = 1, 2, …, U_t and if R2 × C2 > U_t, the dummy bits are padded such that $y_{t,k}$ = 0 or 1 for $k = U_t + 1, U_t + 2, \ldots$, R2 × C2. These dummy bits are pruned away from the output of the matrix after the inter-column permutation.

(4) Perform the inter-column permutation for the matrix based on the pattern $\langle \mathrm{P2}(j) \rangle_{j \in \{0,1,\ldots,\mathrm{C2}-1\}}$ that is shown in table ~~7~~8, where P2(j) is the original column position of the j-th permuted column. After permutation of the columns, the bits are denoted by $y'_{t,k}$.

$$\begin{bmatrix} y'_{t,1} & y'_{t,(\mathrm{R2}+1)} & y'_{t,(2\times\mathrm{R2}+1)} & \cdots & y'_{t,((\mathrm{C2}-1)\times\mathrm{R2}+1)} \\ y'_{t,2} & y'_{t,(\mathrm{R2}+2)} & y'_{t,(2\times\mathrm{R2}+2)} & \cdots & y'_{t,((\mathrm{C2}-1)\times\mathrm{R2}+2)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ y'_{t,\mathrm{R2}} & y'_{t,(2\times\mathrm{R2})} & y'_{t,(3\times\mathrm{R2})} & \cdots & y'_{t,(\mathrm{C2}\times\mathrm{R2})} \end{bmatrix}$$

(5) The output of the block interleaver is the bit sequence read out column by column from the inter-column permuted R2 × C2 matrix. The output is pruned by deleting dummy bits that were padded to the input of the matrix before the inter-column permutation, i.e. bits $y'_{t,k}$ that corresponds to bits $y_{t,k}$ with $k > U_t$ are removed from the output. The bits after time slot 2nd interleaving are denoted by $v_{t,1}, v_{t,2}, \ldots, v_{t,U_t}$, where $v_{t,1}$

corresponds to the bit $y'_{t,k}$ with smallest index k after pruning, $v_{t,2}$ to the bit $y'_{t,k}$ with second smallest index k after pruning, and so on.

Table 8 Inter-column permutation pattern for 2nd interleaving

Number of Columns C2	**Inter-column permutation pattern < P2(0), P2(1), …, P2(C2-1) >**
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>