TSG-RAN WORKING GROUP 1 MEETING NO. 18 *TSGR1-01-0019*

BOSTON, USA, 15-18 JAN. 2001

Agenda Item:	AH99
Source:	Siemens AG
Title:	Bit Scrambling for TDD
Document for:	Approval

---

1 INTRODUCTION

Bit Scrambling for TDD has been proposed at RAN WG1 #17 to overcome the potential problem of DC offsets in specific situations [1]. In the discussion one concern was raised on the usage of Bit Scrambling for the UL.

There are several reasons to use the bit scrambling for the UL as well:

?? In specific conditions (e.g., test, uncoded data like fax), there may be data patterns that result in a DC component for the UL signal as well. Even if the NodeB's receiver may be DC coupled, the transmitter of a simple UE may cancel the DC offset that will result in the same problems as described in [1].

?? The scrambling polynomial and the scrambling function itself will be implemented in the UE anyway for DL, so the savings of not using Bit Scrambling for the UL will consist of only some really marginal increase in computational complexity.

?? Using Bit Scrambling in the UL as well would retain the DL/UL symmetry of the multiplexing chain for TDD.

2 PROPOSAL

With respect to the above mentioned items, we propose to use Bit Scrambling both for the UL and the DL. Corresponding CRs for TS25.221 and TS25.222 are attached to this TDoc.

3 REFERENCES

[1] TDoc R1-00-1340, 'Transmission of Long Sequences with Same Data', Siemens AG, 3GPP TSG RAN WG1#17 meeting, Stockholm, Sweden, 24-27 Nov. 2000

3GPP TSG RAN WG1 Meeting #18
Boston, USA, 15-18 Jan 2001

R1-01-0019

CR-Form-v3

CHANGE REQUEST

✍ **TS25.221** **CR 037** ✍ rev **1** ✍ Current version: **3.5.0** ✍

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 ☒ Radio Access Network ☒ Core Network ☐

Title: ✍	Bit Scrambling for TDD		
Source: ✍	Siemens AG		
Work item code: ✍		***Date:*** ✍	10.1.01
Category: ✍	**F**	***Release:*** ✍	R99

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. This affects the PICH section which has been redrafted.
Consequences if not approved: ✍	Not tolerable restrictions in implementation or not acceptable degradations in the link level performance.

Clauses affected: ✍	5.3.7	
Other specs affected: ✍	☒ Other core specifications ✍ ☐ Test specifications ☐ O&M Specifications	CR222-051r1
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.

5.3.7 The Paging Indicator Channel (PICH)

The Paging Indicator Channel (PICH) is a physical channel used to carry the paging indicators.

5.3.7.1 Mapping of Paging Indicators to the PICH bits

Figure 15 depicts the structure of a PICH burst and the numbering of the bits within the burst. The same burst type is used for the PICH in every cell. N_{PIB} bits in a normal burst of type 1 or 2 are used to carry the paging indicators, where N_{PIB} depends on the burst type: N_{PIB}=240 for burst type 1 and N_{PIB}=272 for burst type 2. The bits ~~b_{NPIB}~~s_{NPIB+1},..., ~~b_{NPIB}~~s_{NPIB+4}~~$_{+3}$~~ adjacent to the midamble are reserved for possible future use.~~ They shall be set to 0 and transmitted with the same power as the paging indicator carrying bits.~~

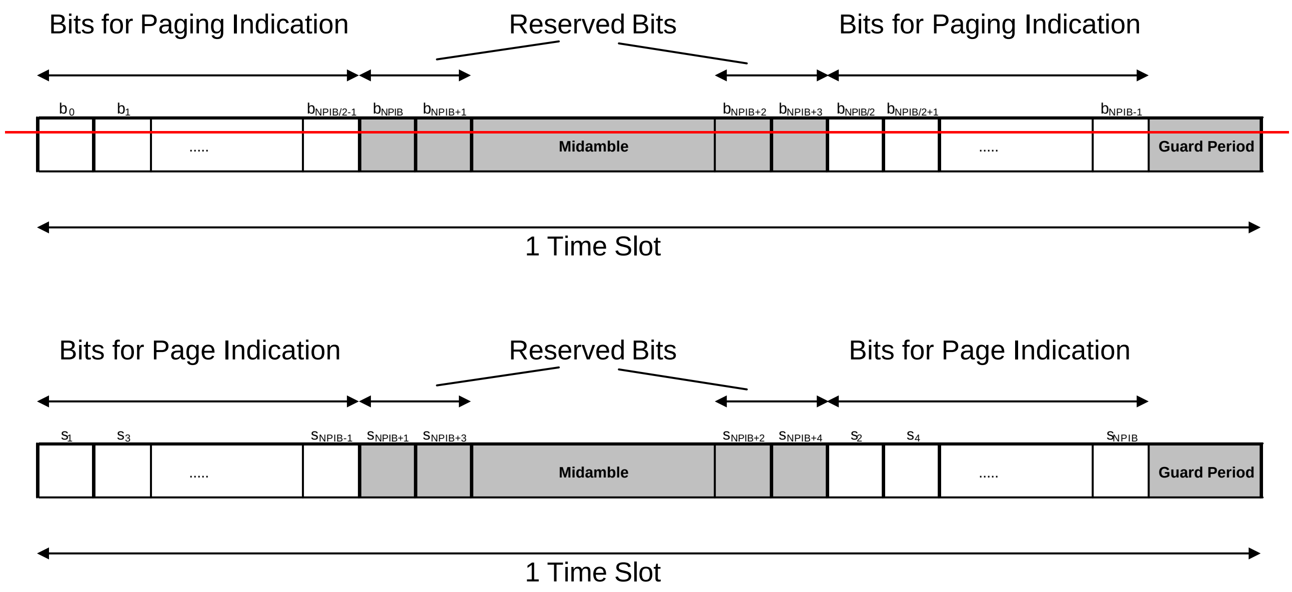


Figure 15: Transmission and numbering of paging indicator carrying bits in a PICH burst

Each paging indicator P_q in one time slot is mapped to the bits {$s_{2Lpi*q+1}$,...,$s_{2Lpi*(q+1)}$} within this time slot. Thus, due to the interleaved transmission of the bits half of the symbols used for each paging indicator are transmitted in the first data part, and the other half of the symbols are transmitted in the second data part, as exemplary shown in figure 17 for a paging indicator length L_{PI} of 4 symbols .

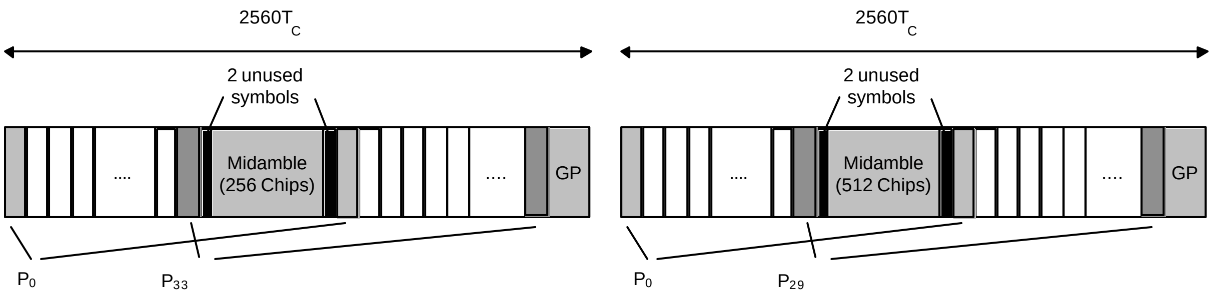


Figure 16: Example of mapping of paging indicators on PICH bits for L_{PI}=4

The setting of the paging indicators and the corresponding PICH bits (including the reserved ones) is described in [7].

In each radio frame~~time slot~~, N_{PI} paging indicators are transmitted, using L_{PI}=2, L_{PI}=4 or L_{PI}=8 symbols.~~ L_{PI} is called the paging indicator length.~~ The number of paging indicators N_{PI} per ~~time slot~~radio frame is given by the paging indicator length and the burst type, which are both known by higher layer signalling. In table 8 this number is shown for the different possibilities of burst types and paging indicator lengths.

Table 8: Number N_{PI} of paging indicators per time slot for the different burst types and paging indicator lengths L_{PI}

	L_{PI}=2	L_{PI}=4	L_{PI}=8
Burst Type 1	N_{PI}=60	N_{PI}=30	N_{PI}=15
Burst Type 2	N_{PI}=68	N_{PI}=34	N_{PI}=17

5.3.7.2 Structure of the PICH over multiple radio frames

As shown in figure 16, the paging indicators of N_{PICH} consecutive frames form a PICH block, N_{PICH} is configured by higher layers. Thus, $N_P=N_{PICH}*N_{PI}$ paging indicators are transmitted in each PICH block.

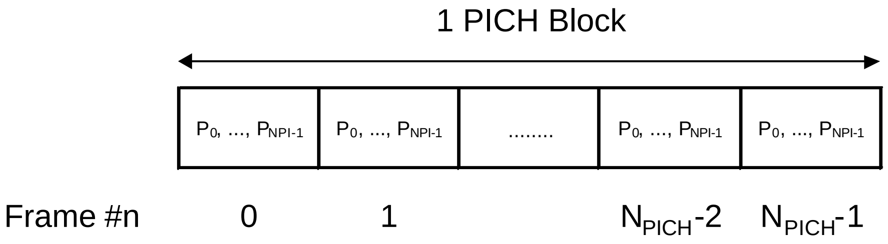


Figure 17~~6~~: Structure of a PICH block

The value PI (PI = 0, ..., N_P-1) calculated by higher layers for use for a certain UE, see [15], is associated to the paging indicator P_q in the nth frame of one PICH block, where q is given by

$q = PI \bmod N_{PI}$

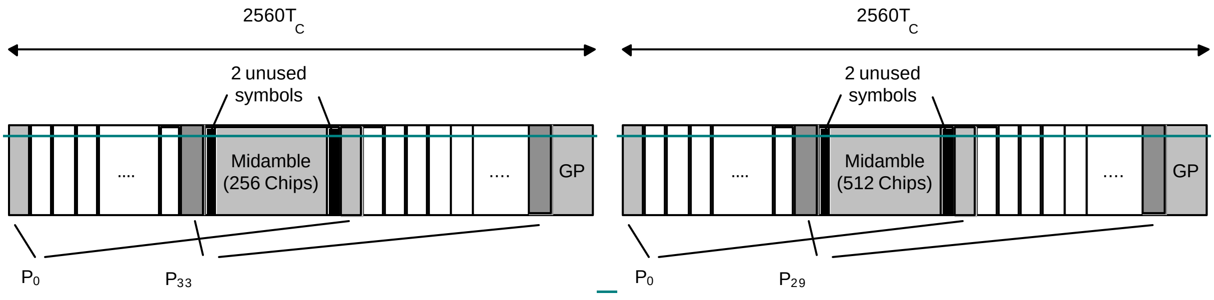
and n is given by

$n = PI \operatorname{div} N_{PI}$.

The PI bitmap in the PCH data frames over Iub contains indication values for all possible higher layer PI values, see [16]. Each bit in the bitmap indicates if the paging indicator P_q associated with that particular PI shall be set to 0 or 1. Hence, the calculation in the formulas above is to be performed in Node B to make the association between PI and P_q.

~~The paging indicator P_q in one time slot is mapped to the bits $\{b_{Lpi*q},\ldots,b_{Lpi*q+Lpi-1},b_{NPIB/2+Lpi*q},\ldots,b_{NPIB/2+Lpi*q+Lpi-1}\}$ within this time slot, as exemplary shown in figure 17. Thus, half of the L_{PI} symbols used for each paging indicator are transmitted in the first data part, and the other half of the L_{PI} symbols are transmitted in the second data part.~~

~~The coding of the paging indicator P_q is given in [7].~~



~~Figure 17: Example of mapping of paging indicators on PICH bits for L_{PI}=4~~

3GPP TSG RAN WG1 Meeting #18
Boston, USA, 15-18 Jan 2001

R1-01-0019

CR-Form-v3

CHANGE REQUEST

✍ **TS25.222** **CR 051** ✍ rev **1** ✍ Current version: **3.5.0** ✍

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 ☐ Radio Access Network ☐ Core Network ☐

Title: ✍	Bit Scrambling for TDD		
Source: ✍	Siemens AG		
Work item code: ✍		***Date:*** ✍	10.1.01
Category: ✍	F	***Release:*** ✍	R99

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.
Consequences if not approved: ✍	Not tolerable restrictions in implementation or not acceptable degradations in the link level performance.

Clauses affected: ✍	4.2; 4.2.8 - 4.2.14; 4.3.2	
Other specs affected: ✍	☒ Other core specifications ☐ Test specifications ☐ O&M Specifications	✍ CR221-037r1
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.

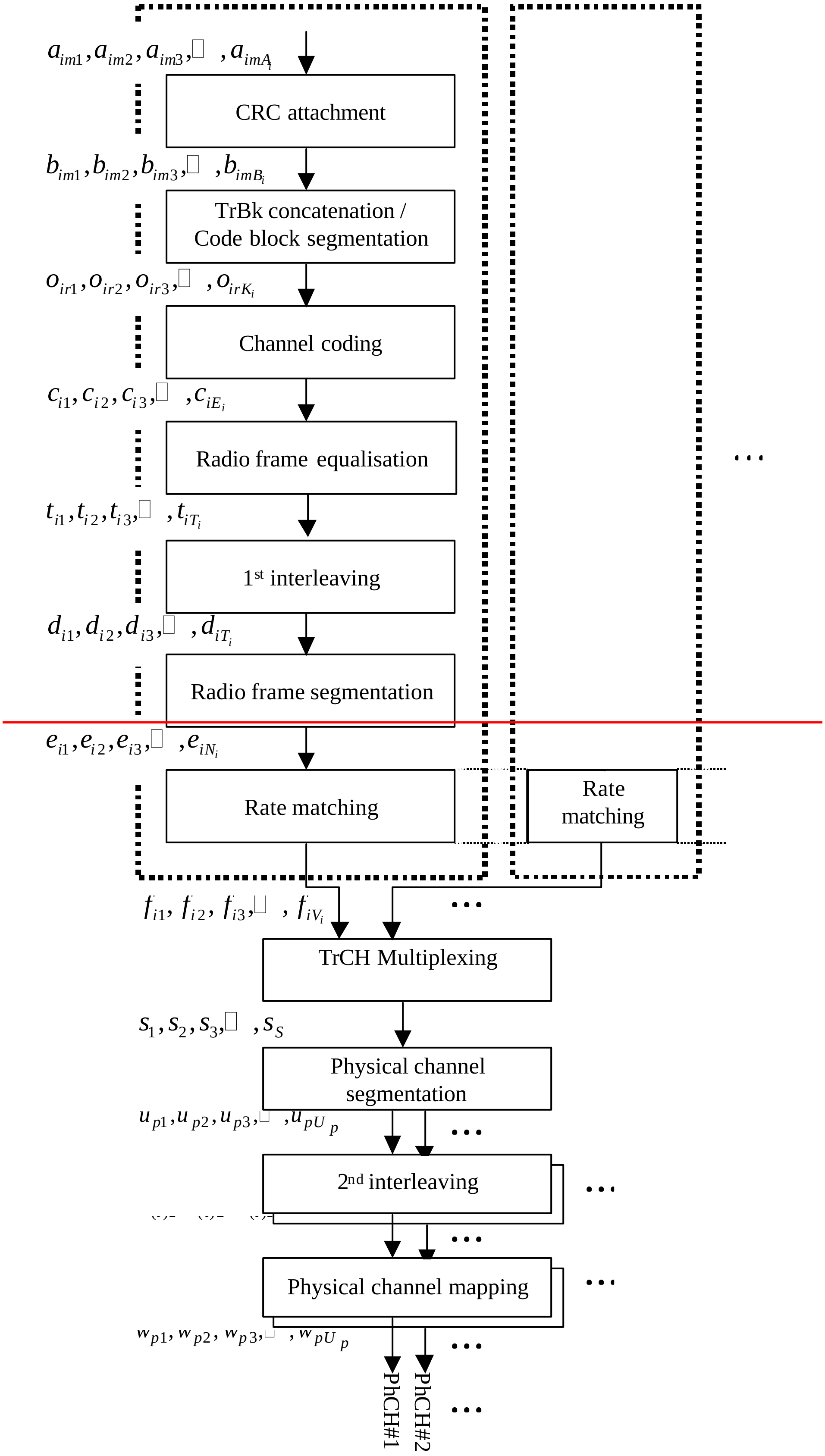
4.2 Transport channel coding/multiplexing

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 {10 ms, 20 ms, 40 ms, 80 ms}.

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~~9~~);
- mapping to physical channels (see subclause 4.2.~~11~~12).

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


$a_{im1}, a_{im2}, a_{im3}, \ldots, a_{imA_i}$
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
…
…
Physical channel mapping
…
$w_{p1}, w_{p2}, w_{p3}, \ldots, w_{pU_p}$
…
PhCH#1
PhCH#2
…

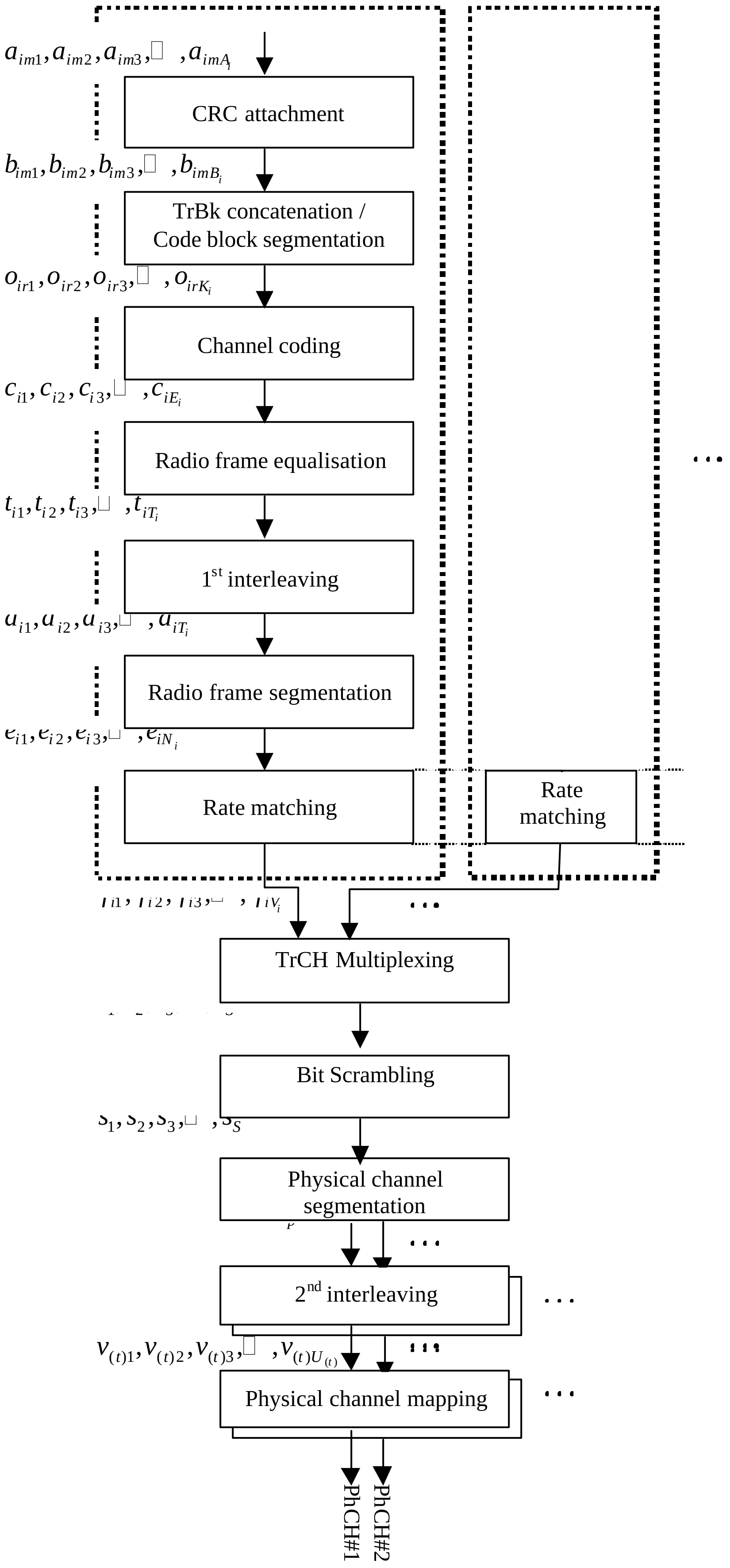


Figure 1: Transport channel multiplexing structure for uplink and downlink

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 figure 1, resulting in several data streams, each mapped to one or several physical channels.

4.2.8 TrCH multiplexing

Every 10 ms, one radio frame from each TrCH is delivered to the TrCH multiplexing. These radio frames are serially multiplexed into a coded composite transport channel (CCTrCH).

The bits input to the TrCH multiplexing are denoted by $f_{i,1}, f_{i,2}, f_{i,3}, \square, f_{i,V_i}$, where *i* is the TrCH number and V_i is the number of bits in the radio frame of TrCH *i*. The number of TrCHs is denoted by *I*. The bits output from TrCH multiplexing are denoted by ~~$s_1, s_2, s_3, \square, s_S$~~ $h_1, h_2, h_3, \square, h_S$, where *S* is the number of bits, i.e. $S ? \underset{i}{?} V_i$. The TrCH multiplexing is defined by the following relations:

~~$s_k ? f_{1,k}$~~ $h_k ? f_{1,k}$ $k = 1, 2, \ldots, V_1$

~~$s_k ? f_{2,(k?V_1)}$~~ $h_k ? f_{2,(k?V_1)}$ $k = V_1+1, V_1+2, \ldots, V_1+V_2$

~~$s_k ? f_{3,(k?(V_1?V_2))}$~~ $h_k ? f_{3,(k?(V_1?V_2))}$ $k = (V_1+V_2)+1, (V_1+V_2)+2, \ldots, (V_1+V_2)+V_3$

□

~~$s_k ? f_{I,(k?(V_1?V_2?\square ?V_{I?1}))}$~~ $h_k ? f_{I,(k?(V_1?V_2?\square ?V_{I?1}))}$ $k = (V_1+V_2+\ldots+V_{I-1})+1, (V_1+V_2+\ldots+V_{I-1})+2, \ldots, (V_1+V_2+\ldots+V_{I-1})+V_I$

4.2.9 Bit Scrambling

The bits output from the TrCH multiplexer are scrambled in the bit scrambler. The bits input to the bit scrambler are denoted by $h_1, h_2, h_3, \square, h_S$, where *S* is the number of bits input to the bit scrambling block equal to the total number of bits on the CCTrCH. The bits after bit scrambling are denoted $s_1, s_2, s_3, \square, s_S$.

Bit scrambling is defined by the following relation:

$s_k ? h_k ? p_k$ $k ? 1,2\square, S$

and p_k results from the following operation:

$p_k ? \underset{i?1}{\overset{16}{??}} g_i ?p_{k?i} ?\mathrm{mod}\ 2$; $p_k ? 0; k ? 1$; $p_1 ? 1$; $g ? ?0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,0,1?$

4.2.10~~9~~ Physical channel segmentation

When more than one PhCH is used, physical channel segmentation divides the bits among the different PhCHs. The bits input to the physical channel segmentation are denoted by $s_1, s_2, s_3, \square, s_S$, where S is the number of bits input to the physical channel segmentation block. The number of PhCHs is denoted by *P*.

The bits after physical channel segmentation are denoted $u_{p,1}, u_{p,2}, u_{p,3}, \square, u_{p,U_p}$, where *p* is PhCH number and U_p is the in general variable number of bits in the respective radio frame for each PhCH. The relation between s_k and $u_{p,k}$ is given below.

Bits on first PhCH after physical channel segmentation:

$u_{1,k} ? s_k$ $k = 1, 2, \ldots, U_1$

Bits on second PhCH after physical channel segmentation:

$u_{2,k} \ ? \ s_{(k?U_1)}$ k = 1, 2 , …, U_2

…

Bits on the P^{th} PhCH after physical channel segmentation:

$u_{P,k} \ ? \ s_{(k?U_1?\square\ ?U_{P?1})}$ k = 1, 2 , …, U_P

4.2.11~~0~~ 2nd interleaving

The 2nd interleaving is a block interleaver and consists of bits input to a matrix with padding, the inter-column permutation for the matrix and bits output from the matrix with pruning. The 2nd interleaving can be applied jointly to all data bits transmitted during one frame, or separately within each timeslot, on which the CCTrCH is mapped. The selection of the 2nd interleaving scheme is controlled by higher layer.

4.2.11~~0~~.1 Frame related 2nd interleaving

In case of frame related 2nd interleaving, the bits input to the block interleaver are denoted by $x_1, x_2, x_3, \square\ , x_U$, where U is the total number of bits after TrCH multiplexing transmitted during the respective radio frame with $S ? U \ ? \underset{p}{?}\ U_p$.

The relation between x_k and the bits $u_{p,k}$ in the respective physical channels is given below:

$x_k \ ? \ u_{1,k}$ k = 1, 2 , …, U_1

$x_{(k?U_1)} \ ? \ u_{2,k}$ k = 1, 2 , …, U_2

…

$x_{(k?U_1?...?U_{P?1})} \ ? \ u_{P,k}$ k = 1, 2 , …, U_P

The following steps have to be performed once for each CCTrCH:

(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 ? R2 X C2.

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

(3) Write the input bit sequence $x_1, x_2, x_3, \square\ , x_U$ into the R2 ? C2 matrix row by row starting with bit y_1 in column 0 of row 0:

$$\begin{array}{ccccccc} ? & y_1 & y_2 & y_3 & \square & y_{C2} & ? \\ ? & y_{(C2?1)} & y_{(C2?2)} & y_{(C2?3)} & \square & y_{(2?C2)} & ? \\ ? & \square & \square & \square & \square & \square & ? \\ ? & y_{((R2?1)?C2?1)} & y_{((R2?1)?C2?2)} & y_{((R2?1)?C2?3)} & \square & y_{(R2?C2)} & ? \end{array}$$

where $y_k \ ? \ x_k$ for k = 1, 2, …, U and if R2 ? C2 > U, the dummy bits are padded such that y_k = 0 or 1 for k = U + 1, U + 2, …, 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, where P$_2$(j) is the original column position of the j-th permuted column. After permutation of the columns, the bits are denoted by y'_k.

$$\begin{bmatrix} y'_1 & y'_{(\mathrm{R2}+1)} & y'_{(2\times\mathrm{R2}+1)} & \cdots & y'_{((\mathrm{C2}-1)\times\mathrm{R2}+1)} \\ y'_2 & y'_{(\mathrm{R2}+2)} & y'_{(2\times\mathrm{R2}+2)} & \cdots & y'_{((\mathrm{C2}-1)\times\mathrm{R2}+2)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ y'_{\mathrm{R2}} & y'_{(2\times\mathrm{R2})} & y'_{(3\times\mathrm{R2})} & \cdots & y'_{(\mathrm{C2}\times\mathrm{R2})} \end{bmatrix}$$

(5) The output of the block interleaver is the bit sequence read out column by column from the inter-column permuted R2 $\times$ 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'_k that corresponds to bits y_k with $k > U$ are removed from the output. The bits after frame related 2nd interleaving are denoted by $v_1, v_2, \ldots, v_U$, where v_1 corresponds to the bit y'_k with smallest index k after pruning, v_2 to the bit y'_k with second smallest index k after pruning, and so on.

4.2.11~~0~~.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:

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

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

…

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

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:

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

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 $\times$ 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 $k = U_t + 1, U_t + 2$, …, 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 $\left\langle \mathrm{P}2?j?\right\rangle_{j?\{0,1,\square\ ,\mathrm{C}2?1?}$ that is shown in table 7, where P2(j) is the original column position of the j-th permuted column. After permutation of the columns, the bits are denoted by $y'_{t,k}$.

$$\begin{array}{ccccccc} ? & y'_{t,1} & y'_{t,(\mathrm{R}2?1)} & y'_{t,(2?\mathrm{R}2?1)} & \square & y'_{t,((\mathrm{C}2\text{-}1)?\mathrm{R}2?1)} & ? \\ ? & y'_{t,2} & y'_{t,(\mathrm{R}2?2)} & y'_{t,(2?\mathrm{R}2?2)} & \square & y'_{t,((\mathrm{C}2\text{-}1)?\mathrm{R}2?2)} & ? \\ ? & \square & \square & \square & \square & \square & ? \\ ? & y'_{t,\mathrm{R}2} & y'_{t,(2?\mathrm{R}2)} & y'_{t,(3?\mathrm{R}2)} & \square & y'_{t,(\mathrm{C}2?\mathrm{R}2)} & ? \end{array}$$

(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 $v_{t,1}, v_{t,2}, \square\ , v_{t,U_t}$, 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 7 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>

4.2.12~~1~~ Physical channel mapping

The PhCH for both uplink and downlink is defined in [6]. The bits after physical channel mapping are denoted by $w_{p,1}, w_{p,2}, \square\ , w_{p,U_p}$, where p is the PhCH number and U_p is the number of bits in one radio frame for the respective PhCH. The bits $w_{p,k}$ are mapped to the PhCHs so that the bits for each PhCH are transmitted over the air in ascending order with respect to k.

The mapping of the bits $v_{(t),1}, v_{(t),2}, \ldots, v_{(t),U_{(t)}}$ is performed like block interleaving, writing the bits into columns, but a PhCH with an odd number is filled in forward order, were as a PhCH with an even number is filled in reverse order.

The mapping scheme, as described in the following subclause, shall be applied individually for each timeslot t used in the current frame. Therefore, the bits $v_{t,1}, v_{t,2}, \ldots, v_{t,U_t}$ are assigned to the bits of the physical channels $w_{t,1,1\ldots U_{t1}}, w_{t,2,1\ldots U_{t2}}, \ldots, w_{t,P_t,1\ldots U_{tP_t}}$ in each timeslot.

In uplink there are at most two codes allocated (P? 2). If there is only one code, the same mapping as for downlink is applied. Denote SF1 and SF2 the spreading factors used for code 1 and 2, respectively. For the number of consecutive bits to assign per code bs_k the following rule is applied:

if

SF1 >= SF2 then $\mathrm{bs}_1 = 1$; bs_2 = SF1/SF2 ;

else

SF2 > SF1 then bs_1 = SF2/SF1; $\mathrm{bs}_2 = 1$;

```
end if
```

In the downlink case bs_p is 1 for all physical channels.

4.2.~~11~~12.1 Mapping scheme

Notation used in this subclause:

P_t: number of physical channels for timeslot t , P_t = 1..2 for uplink ; P_t = 1...16 for downlink

$U_{t,p}$: capacity in bits for the physical channel p in timeslot t

U_t: total number of bits to be assigned for timeslot t

bs_p: number of consecutive bits to assign per code

for downlink all $bs_p = 1$

for uplink if SF1 >= SF2 then $bs_1 = 1$; $bs_2 =$ SF1/SF2 ;

if SF2 > SF1 then $bs_1 =$ SF2/SF1; $bs_2 = 1$;

fb_p: number of already written bits for each code

pos: intermediate calculation variable

```
for  p=1  to $P_t$                     -- reset number of already written bits for every physical channel
    $fb_p = 0$
end for
p = 1                                  -- start with PhCH #1
for k=1  to $U_t$
    do while ($fb_p == U_{t,p}$)          -- physical channel filled up already ?
        p = (p mod $P_t$) + 1;
    end do
    if (p mod 2) == 0
        pos = $U_{t,p} - fb_p$                -- reverse order
    else
        pos = $fb_p + 1$                      -- forward order
    endif
    $w_{t,p,pos} = v_{t,k}$                   -- assignment
    $fb_p = fb_p + 1$                     -- Increment number of already written bits
    if ($fb_p$ mod $bs_p$) == 0             -- Conditional change to the next physical channel
        p = (p mod $P_t$) + 1;
    end if
end for
```

4.2.~~12~~13 Multiplexing of different transport channels onto one CCTrCH, and mapping of one CCTrCH onto physical channels

Different transport channels can be encoded and multiplexed together into one Coded Composite Transport Channel (CCTrCH). The following rules shall apply to the different transport channels which are part of the same CCTrCH:

1) Transport channels multiplexed into one CCTrCh shall have co-ordinated timings. When the TFCS of a CCTrCH is changed because one or more transport channels are added to the CCTrCH or reconfigured within the CCTrCH, or removed from the CCTrCH, the change may only be made at the start of a radio frame with CFN fulfilling the relation

$\text{CFN} \bmod F_{max} = 0$,

where F_{max} denotes the maximum number of radio frames within the transmission time intervals of all transport channels which are multiplexed into the same CCTrCH, including any transport channels i which are added reconfigured or have been removed, and CFN denotes the connection frame number of the first radio frame of the changed CCTrCH.

After addition or reconfiguration of a transport channel i within a CCTrCH, the TTI of transport channel i may only start in radio frames with CFN fulfilling the relation

$\text{CFN}_i \bmod F_i = 0$.

2) Different CCTrCHs cannot be mapped onto the same physical channel.

3) One CCTrCH shall be mapped onto one or several physical channels.

4) Dedicated Transport channels and common transport channels cannot be multiplexed into the same CCTrCH.

5) For the common transport channels, only the FACH and PCH may belong to the same CCTrCH.

6) Each CCTrCH carrying a BCH shall carry only one BCH and shall not carry any other Transport Channel.

7) Each CCTrCH carrying a RACH shall carry only one RACH and shall not carry any other Transport Channel.

Hence, there are two types of CCTrCH.

CCTrCH of dedicated type, corresponding to the result of coding and multiplexing of one or several DCH.

CCTrCH of common type, corresponding to the result of the coding and multiplexing of a common channel, i.e. RACH and USCH in the uplink and DSCH, BCH, FACH or PCH in the downlink, respectively.

Transmission of TFCI is possible for CCTrCH containing Transport Channels of:

- dedicated type;
- USCH type;
- DSCH type;
- FACH and/or PCH type.

4.2.~~12~~13.1 Allowed CCTrCH combinations for one UE

4.2.~~12~~13.1.1 Allowed CCTrCH combinations on the uplink

The following CCTrCH combinations for one UE are allowed, also simultaneously:

1) several CCTrCH of dedicated type;

2) several CCTrCH of common type.

4.2.~~12~~13.1.2 Allowed CCTrCH combinations on the downlink

The following CCTrCH combinations for one UE are allowed, also simultaneously:

3) several CCTrCH of dedicated type;

4) several CCTrCH of common type.

4.2.~~13~~14 Transport format detection

Transport format detection can be performed both with and without Transport Format Combination Indicator (TFCI). If a TFCI is transmitted, the receiver detects the transport format combination from the TFCI. When no TFCI is transmitted, so called blind transport format detection may be used, i.e. the receiver side uses the possible transport format combinations as a priori information.

4.2.~~13~~14.1 Blind transport format detection

Blind Transport Format Detection is optional both in the UE and the UTRAN. Therefore, for all CCTrCH a TFCI shall be transmitted, including the possibilty of a TFCI length zero, if only one TFC is defined.

4.2.~~13~~14.2 Explicit transport format detection based on TFCI

4.2.~~13~~14.2.1 Transport Format Combination Indicator (TFCI)

The Transport Format Combination Indicator (TFCI) informs the receiver of the transport format combination of the CCTrCHs. As soon as the TFCI is detected, the transport format combination, and hence the individual transport channels' transport formats are known, and decoding of the transport channels can be performed.

4.3.2 Coding and Bit Scrambling of the Paging Indicator

The paging indicator P_q-, $q = 0, ..., N_{PI}-1, P_q$? {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 ~~coding~~ of the paging indicators to the bits e_i, $i = 1, ..., N_{PIB}$ is shown in table 10.

Table 10: Mapping ~~Coding~~ of the paging indicator

~~Bits~~	~~Paging Indicator~~	~~Content~~
~~All '0'~~	~~Not set, P_q='0'~~	~~There is no necessity to receive PCH~~
~~All '1'~~	~~Set, P_q='1'~~	~~There is necessity to receive PCH~~

P_q	Bits $\{e_{2Lpi*q+1}, e_{2Lpi*q+2}, ... ,e_{2Lpi*(q+1)}\}$	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].