

# TS 25.222 V2.1.1 1999-09

---

**3<sup>rd</sup> Generation Partnership Project (3GPP);  
Technical Specification Group (TSG)  
Radio Access Network (RAN);  
Working Group 1 (WG1);  
Multiplexing and channel coding (TDD);**

---



Reference

---

<Workitem>

Keywords

---

<keyword[, keyword]>

### **3GPP**

Postal address

---

Office address

---

Internet

---

<http://www.3gpp.org>

---

### **Copyright Notification**

Reproduction is only permitted for the purpose of standardization work undertaken within 3GPP.  
The copyright and the foregoing restrictions extend to reproduction in all media.

© 3GPP 1999.  
All rights reserved.

# Contents

|                         |  |    |
|-------------------------|--|----|
| 1                       | Intellectual Property Rights.....  | 4  |
| 2                       | Foreword.....  | 4  |
| 3                       | Scope.....   | 4  |
| 4                       | References.....  | 4  |
| 5                       | Definitions, symbols and abbreviations .....   | 5  |
| 5.1                     | Definitions.....   | 5  |
| 5.2                     | Symbols.....   | 5  |
| 5.3                     | Abbreviations .....  | 5  |
| 6                       | Multiplexing, channel coding and interleaving .....  | 8  |
| 6.1                     | General.....   | 8  |
| 6.2                     | Transport channel coding/multiplexing .....  | 8  |
| 6.2.1                   | Error detection .....  | 11 |
| 6.2.1.1                 | CRC calculation.....   | 11 |
| 6.2.1.2                 | Relation between input and output of the Cyclic Redundancy Check .....   | 11 |
| 6.2.2                   | Transport block concatenation and code block segmentation .....  | 13 |
| 6.2.2.1                 | Concatenation of transport blocks .....  | 13 |
| 6.2.2.2                 | Code block segmentation .....  | 13 |
| 6.2.3                   | Channel coding .....   | 14 |
| 6.2.3.1                 | Convolutional Coding .....   | 15 |
| 6.2.3.2                 | Turbo coding .....   | 16 |
| 6.2.3.2.1               | Turbo coder .....  | 16 |
| 6.2.3.2.2               | Trellis termination in turbo code.....   | 17 |
| 6.2.3.2.3               | Turbo code internal interleaver.....   | 18 |
| 6.2.4                   | Radio frame size equalisation .....  | 23 |
| 6.2.5                   | 1st interleaving.....  | 23 |
| 6.2.6                   | Radio frame segmentation .....   | 25 |
| 6.2.7                   | Rate matching .....  | 27 |
| 6.2.7.1                 | Determination of rate matching parameters.....   | 27 |
| 6.2.7.2                 | Rate matching algorithm.....   | 28 |
| TrCH multiplexing ..... |  | 29 |
| 6.2.9                   | Physical channel segmentation .....  | 30 |
| 6.2.10                  | 2nd interleaving .....   | 31 |
| 6.2.10.1                | Frame related 2 <sup>nd</sup> interleaving.....  | 31 |
| 6.2.10.2                | Timeslot related 2 <sup>nd</sup> interleaving .....  | 32 |
| 6.2.11                  | Physical channel mapping .....   | 34 |
| 6.2.11.1                | Mapping scheme after frame related 2 <sup>nd</sup> interleaving.....   | 34 |
| 6.2.11.2                | Mapping scheme after timeslot related 2 <sup>nd</sup> interleaving .....   | 34 |
| 6.2.12                  | Multiplexing of different transport channels onto one CCTrCH, and mapping of one CCTrCH onto physical channels ..... | 35 |
| 6.2.12.1                | Allowed CCTrCH combinations for one UE.....  | 36 |
| 6.2.12.1.1              | Allowed CCTrCH combinations on the uplink.....   | 36 |
| 6.2.12.1.2              | Allowed CCTrCH combinations on the downlink .....  | 36 |
| 6.2.13                  | Transport format detection .....   | 36 |
| 6.2.13.1                | Blind transport format detection.....  | 36 |
| 6.2.13.2                | Explicit transport format detection based on TFCI .....  | 36 |
| 6.2.13.2.1              | Transport Format Combination Indicator (TFCI).....   | 36 |
| 6.3                     | Coding for layer 1 control.....  | 37 |
| 6.3.1                   | Coding of transport format combination indicator (TFCI).....   | 37 |
| 6.3.1.1                 | Default TFCI word.....   | 37 |
| 6.3.1.2                 | Coding of short TFCI lengths.....  | 39 |
| 6.3.2                   | Operation of Transport-format-combination indicator (TFCI) in Split Mode.....  | 40 |
| 6.3.3                   | Coding of Paging Indicator (PI).....   | 40 |
| 6.3.4                   | Coding of Transmit Power Control (TPC) .....   | 40 |

|   |               |    |
|---|---------------|----|
| 7 | History ..... | 43 |
|---|---------------|----|

---

## 1 Intellectual Property Rights

*<Editor's note: this section will be completed when an official format for the document is agreed>*

---

## 2 Foreword

This Technical Specification has been produced by the 3GPP.

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of this TS, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version 3.y.z

where:

- x the first digit:
    - 1 presented to TSG for information;
    - 2 presented to TSG for approval;
    - 3 Indicates TSG approved document under change control.
  - y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
  - z the third digit is incremented when editorial only changes have been incorporated in the specification;
- 

## 3 Scope

This 3GPP Report describes multiplexing, channel coding and interleaving for UTRA Physical Layer TDD mode.

Text without revision marks has been approved in the previous TSG-RAN WG1 meetings, while text with revision marks is subject to approval.

---

## 4 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- [1] TS 25.202 (V1.0.0): "UE capabilities"
- [2] TS 25.211 (V1.0.0): "Transport channels and physical channels (FDD)"
- [3] TS 25.212 (V1.0.0): "Multiplexing and channel coding (FDD)"
- [4] TS 25.213 (V1.0.0): "Spreading and modulation (FDD)"
- [5] TS 25.214 (V1.0.0): "Physical layer procedures (FDD)"

- [6] TS 25.221 (V1.0.0): “Transport channels and physical channels (TDD)”
- [7] TS 25.222 (V1.0.0): “Multiplexing and channel coding (TDD)”
- [8] TS 25.223 (V1.0.0): “Spreading and modulation (TDD)”
- [9] TS 25.224 (V1.0.0): “Physical layer procedures (TDD)”
- [10] TS 25.231 (V1.0.0): “Measurements”
- [11] TS S2.01 (V1.0.0): “Radio Interface Protocol Architecture”

## 5 Definitions, symbols and abbreviations

### 5.1 Definitions

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

**<defined term>**: <definition>.

### 5.2 Symbols

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

$\lceil x \rceil$  round towards  $\infty$ , i.e. integer such that  $x \leq \lceil x \rceil < x+1$

$\lfloor x \rfloor$  round towards  $-\infty$ , i.e. integer such that  $x-1 < \lfloor x \rfloor \leq x$

$|x|$  absolute value of  $x$

Unless otherwise is explicitly stated when the symbol is used, the meaning of the following symbols are:

$i$  TrCH number

$j$  TFC number

$k$  Bit number

$l$  TF number

$m$  Transport block number

$n$  Radio frame number

$p$  PhCH number

$r$  Code block number

$I$  Number of TrCHs in a CCTrCH.

$C_i$  Number of code blocks in one TTI of TrCH  $i$ .

$F_i$  Number of radio frames in one TTI of TrCH  $i$ .

$M_i$  Number of transport blocks in one TTI of TrCH  $i$ .

$P$  Number of PhCHs used for one CCTrCH.

$PL$  Puncturing Limit for the uplink. Signalled from higher layers

$RM_i$  Rate Matching attribute for TrCH  $i$ . Signalled from higher layers.

### 5.3 Abbreviations

ARQ                    Automatic Repeat on Request

BCH                    Broadcast Channel

BER                    Bit Error Rate

BPSK                   Binary Phase Shift Keying

|        |  |
|--------|--|
| BS     | Base Station                             |
| BSS    | Base Station Subsystem                   |
| CA     | Capacity Allocation                      |
| CAA    | Capacity Allocation Acknowledgement      |
| CBR    | Constant Bit Rate                        |
| CCCH   | Common Control Channel                   |
| CCTrCH | Coded Composite Transport Channel        |
| CD     | Capacity Deallocation                    |
| CDA    | Capacity Deallocation Acknowledgement    |
| CDMA   | Code Division Multiple Access            |
| CTDMA  | Code Time Division Multiple Access       |
| CRC    | Cyclic Redundancy Check                  |
| DCA    | Dynamic Channel Allocation               |
| DCCH   | Dedicated Control Channel                |
| DL     | Downlink                                 |
| DRX    | Discontinuous Reception                  |
| DSCH   | Downlink Shared Channel                  |
| DTX    | Discontinuous Transmission               |
| FACH   | Forward Access Channel                   |
| FDD    | Frequency Division Duplex                |
| FDMA   | Frequency Division Multiple Access       |
| FEC    | Forward Error Control                    |
| FER    | Frame Error Rate                         |
| GF     | Galois Field                             |
| HCS    | Hierarchical Cell Structure              |
| JD     | Joint Detection                          |
| L1     | Layer 1                                  |
| L2     | Layer 2                                  |
| LLC    | Logical Link Control                     |
| MA     | Multiple Access                          |
| MAC    | Medium Access Control                    |
| MAHO   | Mobile Assisted Handover                 |
| MO     | Mobile Originated                        |
| MOHO   | Mobile Originated Handover               |
| MS     | Mobile Station                           |
| MT     | Mobile Terminated                        |
| NRT    | Non-Real Time                            |
| PC     | Power Control                            |
| PCCC   | Parallel Concatenated Convolutional Code |
| PCH    | Paging Channel                           |
| QoS    | Quality of Service                       |
| QPSK   | Quaternary Phase Shift Keying            |
| RACH   | Random Access Channel                    |
| RF     | Radio Frequency                          |
| RLC    | Radio Link Control                       |
| RRC    | Radio Resource Control                   |
| RRM    | Radio Resource Management                |
| RT     | Real Time                                |

|       |  |
|-------|--|
| RU    | Resource Unit                              |
| SCCC  | Serial Concatenated Convolutional Code     |
| SCH   | Synchronization Channel                    |
| SDCCH | Stand-alone Dedicated Control Channel      |
| SFN   | System Frame Number                        |
| SNR   | Signal to Noise Ratio                      |
| SP    | Switching Point                            |
| TCH   | Traffic channel                            |
| TDD   | Time Division Duplex                       |
| TDMA  | Time Division Multiple Access              |
| TFCI  | Transport Format Combination Indicator     |
| TrBk  | Transport Block                            |
| TrCH  | Transport Channel                          |
| UL    | Uplink                                     |
| UMTS  | Universal Mobile Telecommunications System |
| VBR   | Variable Bit Rate                          |

---

## 6 Multiplexing, channel coding and interleaving

### 6.1 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.

### 6.2 Transport channel coding/multiplexing

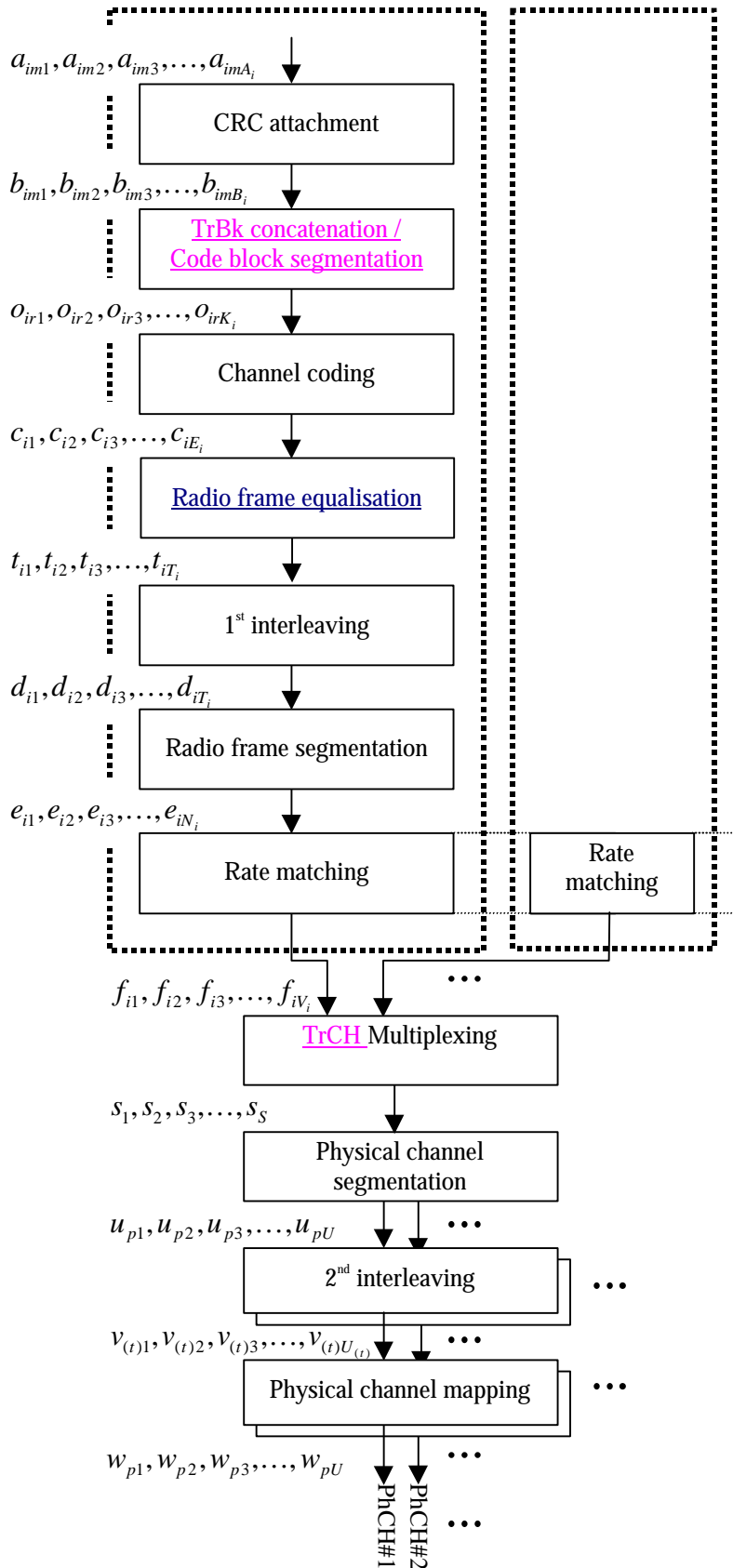
Figure 6-1 illustrates the overall concept of transport-channel coding and multiplexing. Data arrives to the coding/multiplexing unit in form of transport block sets, once every transmission time interval. The transmission time interval is transport-channel specific from the set {10 ms, 20 ms, 40 ms, 80 ms}.

The following coding/multiplexing steps can be identified:

- Add CRC to each transport block (see section 6.2.1)
- TrBk concatenation / Code block segmentation (see section 6.2.2)
- Channel coding (see section 6.2.3)
- Radio frame size equalization (see section 6.2.4)
- Interleaving (two steps, see sections 6.2.5 and 6.2.10)
- Radio frame segmentation (6.2.6)
- Rate matching (see section 6.2.7)
- Multiplexing of transport channels (see section 6.2.8)
- Physical channel segmentation (see section 6.2.9)
- Mapping to physical channels (see section 6.2.11)

The coding/multiplexing steps for uplink and downlink are shown in Figure 6-1.





**Figure 6–1. 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 6-1, resulting in several data streams, each mapped to one or several physical channels.

## 6.2.1 Error detection

Error detection is provided on transport blocks through a Cyclic Redundancy Check. The CRC is 24, 16, 8 or 0 bits and it is signalled from higher layers what CRC length that should be used for each transport channel.

### 6.2.1.1 CRC calculation

The entire transport block is used to calculate the CRC parity bits for each transport block. The parity bits are generated by one of the following cyclic generator polynomials:

$$g_{\text{CRC24}}(D) = D^{24} + D^{23} + D^6 + D^5 + D + 1$$

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

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

Denote the bits in a transport block delivered to layer 1 by  $a_{im1}, a_{im2}, a_{im3}, \dots, a_{imA_i}$ , and the parity bits by

$p_{im1}, p_{im2}, p_{im3}, \dots, p_{imL_i}$ .  $A_i$  is the length of a transport block of TrCH  $i$ ,  $m$  is the transport block number, and  $L_i$  is 24, 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

$$a_{im1}D^{A_i+23} + a_{im2}D^{A_i+22} + \dots + a_{imA_i}D^{24} + p_{im1}D^{23} + p_{im2}D^{22} + \dots + p_{im23}D^1 + p_{im24}$$

yields a remainder equal to 0 when divided by  $g_{\text{CRC24}}(D)$ , polynomial

$$a_{im1}D^{A_i+15} + a_{im2}D^{A_i+14} + \dots + a_{imA_i}D^{16} + p_{im1}D^{15} + p_{im2}D^{14} + \dots + p_{im15}D^1 + p_{im16}$$

yields a remainder equal to 0 when divided by  $g_{\text{CRC16}}(D)$ , and polynomial

$$a_{im1}D^{A_i+7} + a_{im2}D^{A_i+6} + \dots + a_{imA_i}D^8 + p_{im1}D^7 + p_{im2}D^6 + \dots + p_{im7}D^1 + p_{im8}$$

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

### 6.2.1.2 Relation between input and output of the Cyclic Redundancy Check

The bits after CRC attachment are denoted by  $b_{im1}, b_{im2}, b_{im3}, \dots, b_{imB_i}$ , where  $B_i = A_i + L_i$ . The relation between  $a_{imk}$  and  $b_{imk}$  is:

$$b_{imk} = a_{imk} \quad k = 1, 2, 3, \dots, A_i$$

$$b_{imk} = p_{im(L_i+1-(k-A_i))} \quad k = A_i + 1, A_i + 2, A_i + 3, \dots, A_i + L_i$$

## 6.2.2 Transport block concatenation and code block segmentation

All transport blocks in a TTI are serially concatenated. If the number of bits in a TTI is larger than  $Z$ , then code block segmentation is performed after the concatenation of the transport blocks. The maximum size of the code blocks depend on if convolutional or turbo coding is used for the TrCH.

### 6.2.2.1 Concatenation of transport blocks

The bits input to the transport block concatenation are denoted by  $b_{im1}, b_{im2}, b_{im3}, \dots, b_{imB_i}$  where  $i$  is the TrCH number,  $m$  is the transport block number, and  $B_i$  is the number of bits in each block (including CRC). The number of transport blocks on TrCH  $i$  is denoted by  $M_i$ . The bits after concatenation are denoted by  $x_{i1}, x_{i2}, x_{i3}, \dots, x_{iX_i}$ , where  $i$  is the TrCH number and  $X_i = M_i B_i$ . They are defined by the following relations:

$$\begin{aligned}
x_{ik} &= b_{i1k} & k &= 1, 2, \dots, B_i \\
x_{ik} &= b_{i,2,(k-B_i)} & k &= B_i + 1, B_i + 2, \dots, 2B_i \\
x_{ik} &= b_{i,3,(k-2B_i)} & k &= 2B_i + 1, 2B_i + 2, \dots, 3B_i \\
&\dots \\
x_{ik} &= b_{i,M_i,(k-(M_i-1)B_i)} & k &= (M_i - 1)B_i + 1, (M_i - 1)B_i + 2, \dots, M_i B_i
\end{aligned}$$

### 6.2.2.2 Code block segmentation

< Note: It is assumed that filler bits are set to 0 >

Segmentation of the bit sequence from transport block concatenation is performed if  $X_i > Z$ . The code blocks after segmentation are of the same size. The number of code blocks on TrCH  $i$  is denoted by  $C_i$ . If the number of bits input to the segmentation,  $X_i$ , is not a multiple of  $C_i$ , filler bits are added to the last block. The filler bits are transmitted and they are always set to 0. The maximum code block sizes are:

convolutional coding:  $Z = 512 - K_{tail}$

turbo coding:  $Z = 5120 - K_{tail}$

The bits output from code block segmentation are denoted by  $o_{ir1}, o_{ir2}, o_{ir3}, \dots, o_{irK_i}$ , where  $i$  is the TrCH number,  $r$  is the code block number, and  $K_i$  is the number of bits.

Number of code blocks:  $C_i = \lceil X_i / Z \rceil$

Number of bits in each code block:  $K_i = \lceil X_i / C_i \rceil$

Number of filler bits:  $Y_i = C_i K_i - X_i$

If  $X_i \leq Z$ , then  $o_{ik} = x_{ik}$ , and  $K_i = X_i$ .

If  $X_i > Z$ , then

$$\begin{aligned}
o_{1k} &= x_{ik} & k &= 1, 2, \dots, K_i \\
o_{2k} &= x_{i,(k+K_i)} & k &= 1, 2, \dots, K_i \\
o_{3k} &= x_{i,(k+2K_i)} & k &= 1, 2, \dots, K_i \\
&\dots \\
o_{iC_i k} &= x_{i,(k+(C_i-1)K_i)} & k &= 1, 2, \dots, K_i - Y_i \\
o_{iC_i k} &= 0 & k &= (K_i - Y_i) + 1, (K_i - Y_i) + 2, \dots, K_i
\end{aligned}$$

### 6.2.3 Channel coding

Code blocks are delivered to the channel coding block. They are denoted by  $o_{ir1}, o_{ir2}, o_{ir3}, \dots, o_{irK_i}$ , where  $i$  is the TrCH number,  $r$  is the code block number, and  $K_i$  is the number of bits in each code block. The number of code blocks on TrCH  $i$  is denoted by  $C_i$ . After encoding the bits are denoted by  $x_{ir1}, x_{ir2}, x_{ir3}, \dots, x_{irX_i}$ . The encoded blocks are serially multiplexed so that the block with lowest index  $r$  is output first from the channel coding block. The bits output

are denoted by  $c_{i1}, c_{i2}, c_{i3}, \dots, c_{iE_i}$ , where  $i$  is the TrCH number and  $E_i = C_i X_i$ . The output bits are defined by the following relations:

$$c_{ik} = x_{i1k} \quad k = 1, 2, \dots, X_i$$

$$c_{ik} = x_{i,2,(k-X_i)} \quad k = X_i + 1, X_i + 2, \dots, 2X_i$$

$$c_{ik} = x_{i,3,(k-2X_i)} \quad k = 2X_i + 1, 2X_i + 2, \dots, 3X_i$$

...

$$c_{ik} = x_{i,C_i,(k-(C_i-1)X_i)} \quad k = (C_i - 1)X_i + 1, (C_i - 1)X_i + 2, \dots, C_i X_i$$

The relation between  $o_{irk}$  and  $x_{irk}$  and between  $K_i$  and  $X_i$  is dependent on the channel coding scheme.

The following channel coding schemes can be applied to transport channels.

- Convolutional coding
- Turbo coding
- No channel coding

**Table 6.2.3-1 Error Correction Coding Parameters**

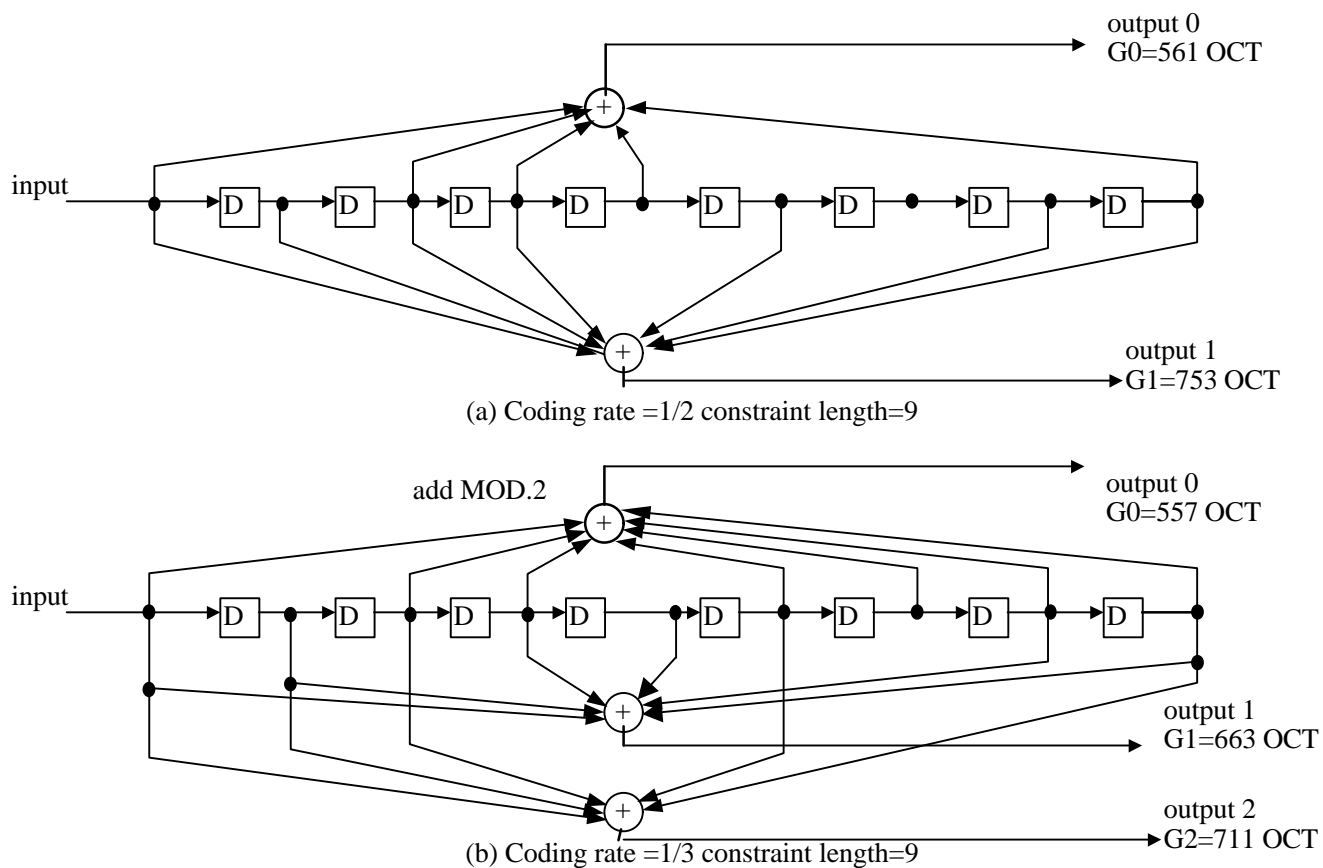
| Transport channel type | Coding scheme      | Coding rate           |
|------------------------|--------------------|-----------------------|
| BCH                    | Convolutional code | 1/2                   |
| PCH                    |                    |                       |
| FACH                   |                    | 1/2                   |
| RACH                   |                    |                       |
| DCH                    | Turbo code         | 1/2, 1/3 or no coding |
| DCH                    |                    |                       |

*Note 1: The exact physical layer encoding/decoding capabilities for different code types are FFS.*

*Note 2: In the UE the channel coding capability should be linked to the terminal class.*

### 6.2.3.1 Convolutional Coding

- Constraint length  $K=9$ . Coding rates 1/2 and 1/3.
- The configuration of the convolutional coder is presented in Fig. 6-2.
- The output from the convolutional coder shall be done in the order output0, output1, output2, output0, output1, ..., output2. (When coding rate is 1/2, output is done up to output 1).
- The initial value of the shift register of the coder shall be "all 0".
- $K-1$  tail bits (value 0) shall be added to the end of the code block before encoding.



**Fig. 6-2 Convolutional Coder**

## 6.2.3.2 Turbo coding

### 6.2.3.2.1 Turbo coder

<Note: It needs to be clarified from TSG SA what are the service specifications with respect to different qualities of service. The performance below BER of  $10^{-6}$  needs to be studied if there is a requirement for this quality of services over the physical layer.>

For data services requiring quality of service between  $10^{-3}$  and  $10^{-6}$  BER inclusive, parallel concatenated convolutional code (PCCC) with 8-state constituent encoders is used.

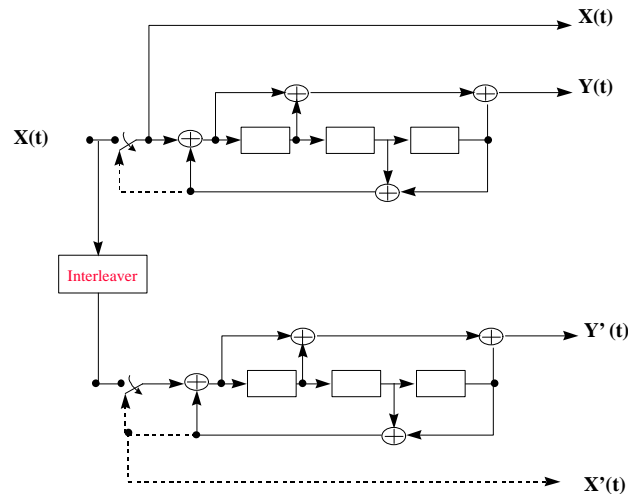
The transfer function of the 8-state constituent code for PCCC is

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

where,

$$d(D) = 1 + D^2 + D^3$$

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



**Figure 6-3. Structure of the 8 state PCCC encoder (dotted lines effective for trellis termination only)**

The initial value of the shift registers of the PCCC encoder shall be all zeros.

The output of the PCCC encoder is punctured to produce coded bits corresponding to the desired code rate 1/3 or 1/2. For rate 1/3, none of the systematic or parity bits are punctured, and the output sequence is  $X(0)$ ,  $Y(0)$ ,  $Y'(0)$ ,  $X(1)$ ,  $Y(1)$ ,  $Y'(1)$ , etc. For rate 1/2, the parity bits produced by the constituent encoders are alternately punctured to produce the output sequence  $X(0)$ ,  $Y(0)$ ,  $X(1)$ ,  $Y'(1)$ ,  $X(2)$ ,  $Y(2)$ ,  $X(3)$ ,  $Y'(3)$ , etc.

### 6.2.3.2.2 Trellis termination in turbo code

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

The first three tail bits shall be used to terminate the first constituent encoder (upper switch of Figure 6-3 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 6-3 in lower position) while the first constituent encoder is disabled.

The transmitted bits for trellis termination shall then be

$$X(t) Y(t) X(t+1) Y(t+1) X(t+2) Y(t+2) X'(t) Y'(t) X'(t+1) Y'(t+1) X'(t+2) Y'(t+2).$$

### 6.2.3.2.3 Turbo code internal interleaver

Figure 6-5 depicts the overall 8 State PCCC Turbo coding scheme including Turbo code internal interleaver. The Turbo code internal interleaver consists of mother interleaver generation and pruning. For arbitrary given block length  $K$ , one mother interleaver is selected from the 134 mother interleavers set. The generation scheme of mother interleaver is described in section 6.2.2.2.3.1. After the mother interleaver generation,  $l$ -bits are pruned in order to adjust the mother interleaver to the block length  $K$ . The definition of  $l$  is shown in section 6.2.3.2.3.2..

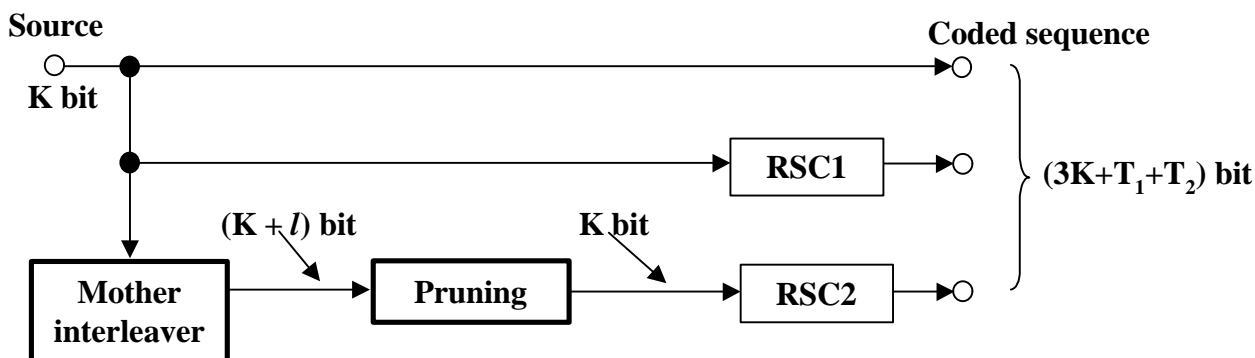


Figure 6-4. Overall 8 State PCCC Turbo Coding

#### 6.2.3.2.3.1 Mother interleaver generation

The interleaving consists of three stages. In first stage, the input sequence is written into the rectangular matrix row by row. The second stage is intra-row permutation. The third stage is inter-row permutation. The three-stage permutations are described as follows, the input block length is assumed to be  $K$  (320 to 5114 bits).

##### First Stage:

- (1) Determine a row number  $R$  such that

$$R=10 \text{ (} K = 481 \text{ to } 530 \text{ bits; Case-1)}$$

$$R=20 \text{ (} K = \text{any other block length except } 481 \text{ to } 530 \text{ bits; Case-2)}$$

- (2) Determine a column number  $C$  such that

$$\text{Case-1; } C = p = 53$$

Case-2;

- (i) find minimum prime  $p$  such that,

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

- (ii) if  $(0 \leq p-K/R)$  then go to (iii)

$$\text{else } C = p+1.$$

- (iii) if  $(0 \leq p-1-K/R)$  then  $C=p-1$ .

$$\text{else } C = p.$$

- (3) The input sequence of the interleaver is written into the  $R \times C$  rectangular matrix row by row.

##### Second Stage:

###### A. If $C = p$

- (A-1) Select a primitive root  $g_0$  from Table 6.2.2-2.

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

$$c(i) = [g_0 \times c(i-1)] \bmod p, i = 1, 2, \dots, (p-2), c(0) = 1.$$

(A-3) Select the minimum prime integer set  $\{q_j\}$  ( $j=1,2,\dots,R-1$ ) such that

$$\text{g.c.d}\{q_j, p-1\} = 1$$

$$q_j > 6$$

$$q_j > q_{(j-1)}$$

where g.c.d. is greatest common divider. And  $q_0 = 1$ .

(A-4) The set  $\{q_j\}$  is permuted to make a new set  $\{p_j\}$  such that

$$p_{P(j)} = q_j, \quad j = 0, 1, \dots, R-1,$$

where  $P(j)$  is the inter-row permutation pattern defined in the third stage.

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

$$c_j(i) = c([i \times p_j] \bmod (p-1)), \quad i = 0, 1, 2, \dots, (p-2), \text{, and } c_j(p-1) = 0,$$

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

B. If  $C = p+1$ (B-1) Same as case A-1.

(B-2) Same as case A-2.

(B-3) Same as case A-3.

(B-4) Same as case A-4.

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

$$c_j(i) = c([i \times p_j] \bmod (p-1)), \quad i = 0, 1, 2, \dots, (p-2), \text{, } c_j(p-1) = 0, \text{ and } c_j(p) = p,$$

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

(B-6) If ( $K = C \times R$ ) then exchange  $c_{R-1}(p)$  with  $c_{R-1}(0)$ .

C. If  $C = p-1$

(C-1) Same as case A-1.

(C-2) Same as case A-2.

(C-3) Same as case A-3.

(C-4) Same as case A-4.

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

$$c_j(i) = c([i \times p_j] \bmod (p-1)) - 1, \quad i = 0, 1, 2, \dots, (p-2), \text{,}$$

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

**Third Stage:**





|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

#### 6.2.3.2.3.2 Definition of the number of pruning bits

The output of the mother interleaver is pruned by deleting the  $l$ -bits in order to adjust the mother interleaver to the block length  $K$ , where the deleted bits are non-existent bits in the input sequence. The pruning bits number  $l$  is defined as:

$$l = R \times C - K,$$

where  $R$  is the row number and  $C$  is the column number defined in section 6.2.3.2.3.1.

### 6.2.4 Radio frame size equalisation

Radio frame size equalisation is padding the input bit sequence in order to ensure that the output can be segmented in  $F_i$  data segments of same size as described in the section 6.2.6.

The input bit sequence to the radio frame size equalisation is denoted by  $c_{i1}, c_{i2}, c_{i3}, \dots, c_{iE_i}$ , where  $i$  is TrCH number and  $E_i$  the number of bits. The output bit sequence is denoted by  $t_{i1}, t_{i2}, t_{i3}, \dots, t_{iT_i}$ , where  $T_i$  is the number of bits. The output bit sequence is derived as follows:

$$t_{ik} = c_{ik}, \text{ for } k = 1 \dots E_i \text{ and}$$

$$t_{ik} = \{0 \mid 1\} \text{ for } k = E_i + 1 \dots T_i, \text{ if } E_i < T_i$$

where

$$T_i = F_i * N_i \text{ and}$$

$$N_i = \left\lceil \frac{(E_i - 1)}{F_i} \right\rceil + 1 \text{ is the number of bits per segment after size equalisation.}$$

### 6.2.5 1st interleaving

The 1<sup>st</sup> interleaving is a block interleaver with inter-column permutations. The input bit sequence to the 1<sup>st</sup> interleaver is denoted by  $x_{i1}, x_{i2}, x_{i3}, \dots, x_{iX_i}$ , where  $i$  is TrCH number and  $X_i$  the number of bits (at this stage  $X_i$  is assumed and guaranteed to be an integer multiple of TTI). The output bit sequence is derived as follows:

- (1) Select the number of columns  $C_l$  from Table 6.2.5-1.
- (2) Determine the number of rows  $R_l$  defined as  $R_l = X_i / C_l$

- (3) Write the input bit sequence into the  $R_I \times C_I$  rectangular matrix row by row starting with bit  $x_{i,1}$  in the first column of the first row and ending with bit  $x_{i,(R_I C_I)}$  in column  $C_I$  of row  $R_I$ :

$$\begin{bmatrix} x_{i1} & x_{i2} & x_{i3} & \dots & x_{iC_I} \\ x_{i,(C_I+1)} & x_{i,(C_I+2)} & x_{i,(C_I+3)} & \dots & x_{i,(2C_I)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ x_{i,((R_I-1)C_I+1)} & x_{i,((R_I-1)C_I+2)} & x_{i,((R_I-1)C_I+3)} & \dots & x_{i,(R_I C_I)} \end{bmatrix}$$

- (4) Perform the inter-column permutation based on the pattern  $\{P_1(j)\}$  ( $j=0,1, \dots, C_I-1$ ) shown in Table 4-3, where  $P_1(j)$  is the original column position of the  $j$ -th permuted column. After permutation of the columns, the bits are denoted by  $y_{ik}$ :

$$\begin{bmatrix} y_{i1} & y_{i,(R_I+1)} & y_{i,(2R_I+1)} & \dots & y_{i,((C_I-1)R_I+1)} \\ y_{i2} & y_{i,(R_I+2)} & y_{i,(2R_I+2)} & \dots & y_{i,((C_I-1)R_I+2)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ y_{iR_I} & y_{i,(2R_I)} & y_{i,(3R_I)} & \dots & y_{i,(C_I R_I)} \end{bmatrix}$$

- (5) Read the output bit sequence  $y_{i1}, y_{i2}, y_{i3}, \dots, y_{i,(C_I R_I)}$  of the 1<sup>st</sup> interleaving column by column from the inter-column permuted  $R_I \times C_I$  matrix. Bit  $y_{i,1}$  corresponds to the first row of the first column and bit  $y_{i,(R_I C_I)}$  corresponds to row  $R_I$  of column  $C_I$ .

The bits input to the 1<sup>st</sup> interleaving are denoted by  $t_{i1}, t_{i2}, t_{i3}, \dots, t_{iT_i}$ , where  $i$  is the TrCH number and  $E_i$  the number of bits. Hence,  $x_{ik} = t_{ik}$  and  $X_i = T_i$ .

The bits output from the 1<sup>st</sup> interleaving are denoted by  $d_{i1}, d_{i2}, d_{i3}, \dots, d_{iT_i}$ , and  $d_{ik} = y_{ik}$ .

**Table 6.2.5-1**

| TTI   | Number of columns $C_I$ | Inter-column permutation patterns |
|-------|-------------------------|-----------------------------------|
| 10 ms | 1                       | {0}                               |
| 20 ms | 2                       | {0,1}                             |
| 40 ms | 4                       | {0,2,1,3}                         |
| 80 ms | 8                       | {0,4,2,6,1,5,3,7}                 |

## 6.2.6 Radio frame segmentation

When the transmission time interval is longer than 10 ms, the input bit sequence is segmented and mapped onto consecutive radio frames. Following radio frame size equalisation the input bit sequence length is guaranteed to be an integer multiple of  $F_i$ .

The input bit sequence is denoted by  $x_{i1}, x_{i2}, x_{i3}, \dots, x_{iX_i}$  where  $i$  is the TrCH number and  $X_i$  is the number bits. The  $F_i$  output bit sequences per TTI are denoted by  $y_{i,n_i1}, y_{i,n_i2}, y_{i,n_i3}, \dots, y_{i,n_iY_i}$  where  $n_i$  is the radio frame number in current TTI and  $Y_i$  is the number of bits per radio frame for TrCH  $i$ . The output sequences are defined as follows:

$$y_{i,n_i,k} = x_{i,((n_i-1)Y_i)+k}, n_i = 1 \dots F_i, j = 1 \dots Y_i$$

where

$Y_i = (X_i / F_i)$  is the number of bits per segment,

$x_{ik}$  is the  $k^{\text{th}}$  bit of the input bit sequence and

$y_{i,n_i,k}$  is the  $k^{\text{th}}$  bit of the output bit sequence corresponding to the  $n^{\text{th}}$  radio frame

The  $n_i$ -th segment is mapped to the  $n_i$ -th radio frame of the transmission time interval.

The input bit sequence to the radio frame segmentation is denoted by  $d_{i1}, d_{i2}, d_{i3}, \dots, d_{iT_i}$ , where  $i$  is the TrCH number and  $T_i$  the number of bits. Hence,  $x_{ik} = d_{ik}$  and  $X_i = T_i$ .

The output bit sequence corresponding radio frame  $n_i$  is denoted by  $e_{i1}, e_{i2}, e_{i3}, \dots, e_{iN_i}$ , where  $i$  is the TrCH number and  $N_i$  is the number of bits. Hence,  $e_{i,k} = y_{i,n_i,k}$  and  $N_i = Y_i$ .

## 6.2.7 Rate matching

Rate matching means that bits on a TrCH are repeated or punctured. Higher layers assign a rate-matching attribute for each TrCH. This attribute is semi-static and can only be changed through higher layer signalling. The rate-matching attribute is used when the number of bits to be repeated or punctured is calculated.

The number of bits on a TrCH can vary between different transmission time intervals. When the number of bits between different transmission time intervals is changed, bits are repeated to ensure that the total bit rate after second multiplexing is identical to the total channel bit rate of the allocated dedicated physical channels.

**Notation used in Section 6.2.7 and subsections:**

- $N_{ij}$ : Number of bits in a radio frame before rate matching on TrCH  $i$  with transport format combination  $j$ .
- $\Delta N_{ij}$ : If positive - number of bits to be repeated in each radio frame on TrCH  $i$  with transport format combination  $j$ .  
If negative - number of bits to be punctured in each radio frame on TrCH  $i$  with transport format combination  $j$ .
- $RM_i$ : Semi-static rate matching attribute for TrCH  $i$ . Signalled from higher layers.
- $PL$ : Puncturing limit for uplink. This value limits the amount of puncturing that can be applied in order to minimise the number of dedicated physical channels. Signalled from higher layers.
- $N_{data,j}$ : Total number of bits that are available for a CCTrCH in a radio frame with transport format combination  $j$ .
- $I$ : Number of TrCHs in a CCTrCH.
- $Z_{mj}$ : Intermediate calculation variable.
- $F_i$ : Number of radio frames in the transmission time interval of TrCH  $i$ .
- $n_i$ : Radio frame number in the transmission time interval of TrCH  $i$  ( $0 \leq n_i < F_i$ ).
- $q$ : Average puncturing distance.

$I_F(n_i)$ : The inverse interleaving function of the 1<sup>st</sup> interleaver (note that the inverse interleaving function is identical to the interleaving function itself for the 1<sup>st</sup> interleaver).

$S(n_i)$ : The shift of the puncturing pattern for radio frame  $n_i$ .

$TF_i(j)$ : Transport format of TrCH  $i$  for the transport format combination  $j$ .

### 6.2.7.1 Determination of rate matching parameters

The following relations are used when calculating the rate matching pattern:

$$Z_{0,j} = 0$$

$$Z_{mj} = \left[ \frac{\sum_{i=1}^m RM_i \cdot N_{ij}}{\sum_{i=1}^T RM_i \cdot N_{ij}} \cdot N_{data,j} \right] \text{ for all } m = 1 \dots l$$

$$\Delta N_{ij} = Z_{ij} - Z_{i-1,j} - N_{ij} \quad \text{for all } i = 1 \dots l$$

Puncturing can be used to minimise the number of required transmission capacity. The maximum amount of puncturing that can be applied is signalled from higher layers and denoted by  $PL$ . The possible values for  $N_{data}$  in uplink and downlink depend on the dedicated physical channels which are assigned to the link, respectively. The supported set of  $N_{data}$ , denoted SET0, depends on the UE capabilities.  $N_{data,j}$  for the transport format combination  $j$  is determined by executing the following algorithm:

$$SET1 = \{ N_{data} \text{ in SET0 such that } N_{data} - PL \cdot \sum_{i=1}^T \frac{RM_i}{\min_l \{RM_l\}} \cdot N_{ij} \text{ is non negative } \}$$

$$N_{data,j} = \min SET1$$

The number of bits to be repeated or punctured,  $\mathbf{DN}_{ij}$ , within one radio frame for each TrCH  $i$  is calculated with the relations given at the beginning of this section for all possible transport format combinations  $j$  and selected every radio frame. For each radio frame, the rate-matching pattern is calculated with the algorithm in Section 6.2.6.2, where  $\mathbf{DN} = \mathbf{DN}_{ij}$  and  $N = N_{ij}$ .

Additionally, the following parameters are needed:

$$q = \lceil N_{ij} / (\hat{\mathbf{O}} \mathbf{DN}_{ij} \hat{\mathbf{O}}) \rceil$$

if  $q$  is even

$$\text{then } q' = q - \text{gcd}(q, F_i) / F_i \text{ -- where } \text{gcd}(q, F_i) \text{ means greatest common divisor of } q \text{ and } F_i$$

-- note that  $q'$  is not an integer, but a multiple of 1/8

else

$$q' = q$$

endif

for  $l = 0$  to  $F_i - 1$

$$S(I_F(\hat{\mathbf{e}} * q' \hat{\mathbf{u}} \bmod F_i)) = (\hat{\mathbf{e}} * q' \hat{\mathbf{u}} \text{ div } F_i)$$

end for

### 6.2.7.2 Rate matching algorithm

The bits input to the rate matching are denoted by  $e_{i1}, e_{i2}, e_{i3}, \dots, e_{iN_i}$ , where  $i$  is the TrCH with  $N = N_{ij} = N_i$ .

The bits output from the rate matching are denoted by  $f_{i1}, f_{i2}, f_{i3}, \dots, f_{iV_i}$ , where  $i$  is the TrCH number and  $V_i = N + DN = N_{ij} + DN_{ij}$ .

Note that the transport format combination number  $j$  for simplicity has been left out in the bit numbering.

The rate matching rule is as follows:

*if puncturing is to be performed*

```

y = -DN
e = (2*S(ni) * y + N) mod 2N -- initial error between current and desired puncturing ratio
m = 1 -- index of current bit
do while m <= N
  e = e - 2 * y -- update error
  if e <= 0 then -- check if bit number m should be punctured
    puncture bit xm
    e = e + 2*N -- update error
  end if
  m = m + 1 -- next bit
end do

```

*else*

```

y = DN
e = (2*S(ni) * y + N) mod 2N -- initial error between current and desired puncturing ratio
m = 1 -- index of current bit
do while m <= N
  e = e - 2 * y -- update error
  do while e <= 0 -- check if bit number m should be repeated
    repeat bit xm
    e = e + 2*N -- update error
  enddo
  m = m + 1 -- next bit
end do

```

*end if*

A repeated bit is placed directly after the original one.

## 6.2.8 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  $f_{i1}, f_{i2}, f_{i3}, \dots, f_{iV_i}$ , 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  $s_1, s_2, s_3, \dots, s_S$ , where  $S$  is the number of bits, i.e.  $S = \sum_i V_i$ . The TrCH multiplexing

is defined by the following relations:

$$\begin{aligned}
 s_k &= f_{1k} & k &= 1, 2, \dots, V_1 \\
 s_k &= f_{2,(k-V_1)} & k &= V_1+1, V_1+2, \dots, V_1+V_2 \\
 s_k &= f_{3,(k-(V_1+V_2))} & k &= (V_1+V_2)+1, (V_1+V_2)+2, \dots, (V_1+V_2)+V_3 \\
 \dots & & & \\
 s_k &= f_{I,(k-(V_1+V_2+\dots+V_{I-1}))} & k &= (V_1+V_2+\dots+V_{I-1})+1, (V_1+V_2+\dots+V_{I-1})+2, \dots, (V_1+V_2+\dots+V_{I-1})+V_I
 \end{aligned}$$

## 6.2.9 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  $s_1, s_2, s_3, \dots, s_S$ , 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  $u_{p1}, u_{p2}, u_{p3}, \dots, u_{pU_p}$ , 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  $u_{pk}$  is given below.

Bits on first PhCH after physical channel segmentation:

$$u_{1k} = s_k \quad k = 1, 2, \dots, U_1$$

Bits on second PhCH after physical channel segmentation:

$$u_{2k} = s_{(k+U_1)} \quad k = 1, 2, \dots, U_2$$

...

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

$$u_{Pk} = s_{(k+U_1+\dots+U_{p-1})} \quad k = 1, 2, \dots, U_P$$

## 6.2.10 2nd interleaving

The 2<sup>nd</sup> 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 2<sup>nd</sup> interleaving scheme is controlled by higher layer.

### 6.2.10.1 Frame related 2<sup>nd</sup> interleaving

In case of frame related interleaving, the bits input to the 2<sup>nd</sup> interleaver are denoted  $x_1, x_2, x_3, \dots, x_U$ , where  $U$  is the total number of bits after TrCH multiplexing transmitted during the respective radio frame.

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

$$x_k = u_{1k} \quad k = 1, 2, \dots, U_1$$

$$x_{(k+U_1)} = u_{2k} \quad k = 1, 2, \dots, U_2$$

...

$$x_{(k+U_1+\dots+U_{p-1})} = u_{Pk} \quad k = 1, 2, \dots, U_P$$

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

(1) Set the number of columns  $C_2 = 30$ . The columns are numbered  $0, 1, 2, \dots, C_2-1$  from left to right.

(2) Determine the number of rows  $R_2$  by finding minimum integer  $R_2$  such that  $U \leq R_2 C_2$ .

(3) The bits input to the 2<sup>nd</sup> interleaving are written into the  $R_2 \times C_2$  rectangular matrix row by row.

$$\begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_{30} \\ x_{31} & x_{32} & x_{33} & \dots & x_{60} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ x_{(R_2-1)30+1} & x_{(R_2-1)30+2} & x_{(R_2-1)30+3} & \dots & x_{R_2 \cdot 30} \end{bmatrix}$$

(4) Perform the inter-column permutation based on the pattern  $\{P_2(j)\}$  ( $j = 0, 1, \dots, C_2-1$ ) that is shown in Table 6.2.9-1, 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  $y_k$ .

$$\begin{bmatrix} y_1 & y_{R_2+1} & y_{2R_2+1} & \cdots & y_{29R_2+1} \\ y_2 & y_{R_2+2} & y_{2R_2+2} & \cdots & y_{29R_2+2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ y_{R_2} & y_{2R_2} & y_{3R_2} & \cdots & y_{30R_2} \end{bmatrix}$$

- (5) The output of the 2<sup>nd</sup> interleaving is the bit sequence read out column by column from the inter-column permuted  $R_2 \times C_2$  matrix. 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  $k > U$  are removed from the output. The bits after 2<sup>nd</sup> interleaving are denoted by  $v_1, v_2, \dots, v_U$ , 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.

### 6.2.10.2 Timeslot related 2<sup>nd</sup> interleaving

In case of timeslot related 2<sup>nd</sup> interleaving, the bits input to the 2<sup>nd</sup> interleaver are denoted  $x_{t1}, x_{t2}, x_{t3}, \dots, x_{tU_t}$ , 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  $x_{tk}$  and  $u_{pk}$  is given below with  $P_t$  referring to the number of physical channels within the respective timeslot:

$$x_{tk} = u_{1k} \quad k = 1, 2, \dots, U_1$$

$$x_{t(k+U_1)} = u_{2k} \quad k = 1, 2, \dots, U_2$$

...

$$x_{t(k+U_1+\dots+U_{P_t-1})} = u_{P_t k} \quad k = 1, 2, \dots, U_{P_t}$$

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

- (1) Set the number of columns  $C_2 = 30$ . The columns are numbered 0, 1, 2, ...,  $C_2-1$  from left to right.
- (2) Determine the number of rows  $R_2$  by finding minimum integer  $R_2$  such that  $U_t \leq R_2 C_2$ .
- (3) The bits input to the 2<sup>nd</sup> interleaving are written into the  $R_2 \times C_2$  rectangular matrix row by row.

$$\begin{bmatrix} x_{t1} & x_{t2} & x_{t3} & \cdots & x_{t30} \\ x_{t31} & x_{t32} & x_{t33} & \cdots & x_{t60} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ x_{t,((R_2-1)30+1)} & x_{t,((R_2-1)30+2)} & x_{t,((R_2-1)30+3)} & \cdots & x_{t,(R_2 30)} \end{bmatrix}$$

- (4) Perform the inter-column permutation based on the pattern  $\{P_2(j)\}$  ( $j = 0, 1, \dots, C_2-1$ ) that is shown in Table 6.2.9-1, 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  $y_{tk}$ .

$$\begin{bmatrix} y_{t1} & y_{t,(R_2+1)} & y_{t,(2R_2+1)} & \cdots & y_{t,(29R_2+1)} \\ y_{t2} & y_{t,(R_2+2)} & y_{t,(2R_2+2)} & \cdots & y_{t,(29R_2+2)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ y_{tR_2} & y_{t,(2R_2)} & y_{t,(3R_2)} & \cdots & y_{t,(30R_2)} \end{bmatrix}$$

- (5) The output of the 2<sup>nd</sup> interleaving is the bit sequence read out column by column from the inter-column permuted  $R_2 \times C_2$  matrix. The output is pruned by deleting bits that were not present in the input bit sequence, i.e. bits  $y_{tk}$  that corresponds to bits  $x_{tk}$  with  $k > U_t$  are removed from the output. The bits after 2<sup>nd</sup> interleaving are denoted by



$v_{i1}, v_{i2}, \dots, v_{iU_i}$ , where  $v_{i1}$  corresponds to the bit  $y_{ik}$  with smallest index  $k$  after pruning,  $v_{i2}$  to the bit  $y_{ik}$  with second smallest index  $k$  after pruning, and so on.

Table 6.2.10-1

| Column number $C_2$ | 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} |

## 6.2.11 Physical channel mapping

The PhCH for both uplink and downlink is defined in [6]. The bits after physical channel mapping are denoted by  $w_{p1}, w_{p2}, \dots, w_{pU_p}$ , where  $p$  is the PhCH number and  $U_p$  is the number of bits in one radio frame for the respective PhCH. The bits  $w_{pk}$  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 scheme depends on the applied  $2^{\text{nd}}$  interleaving scheme.

### 6.2.11.1 Mapping scheme after frame related $2^{\text{nd}}$ interleaving

The following mapping rule is applied:

Bits on first PhCH after physical channel mapping:

$$w_{1k} = v_k \quad k = 1, 2, \dots, U_1$$

Bits on second PhCH after physical channel mapping:

$$w_{2k} = v_{(k+U_1)} \quad k = 1, 2, \dots, U_2$$

...

Bits on the  $P^{\text{th}}$  PhCH after physical channel mapping:

$$w_{Pk} = v_{(k+U_1+\dots+U_{p-1})} \quad k = 1, 2, \dots, U_P$$

### 6.2.11.2 Mapping scheme after timeslot related $2^{\text{nd}}$ interleaving

For each timeslot only those physical channels with  $p = 1, 2, \dots, P_t$  are considered respectively, which are transmitted in that timeslot, and the following mapping scheme is applied:

Bits on first PhCH in timeslot  $t$  after physical channel mapping:

$$w_{1k} = v_{tk} \quad k = 1, 2, \dots, U_1$$

Bits on second PhCH in timeslot  $t$  after physical channel mapping:

$$w_{2k} = v_{t(k+U_1)} \quad k = 1, 2, \dots, U_2$$

...

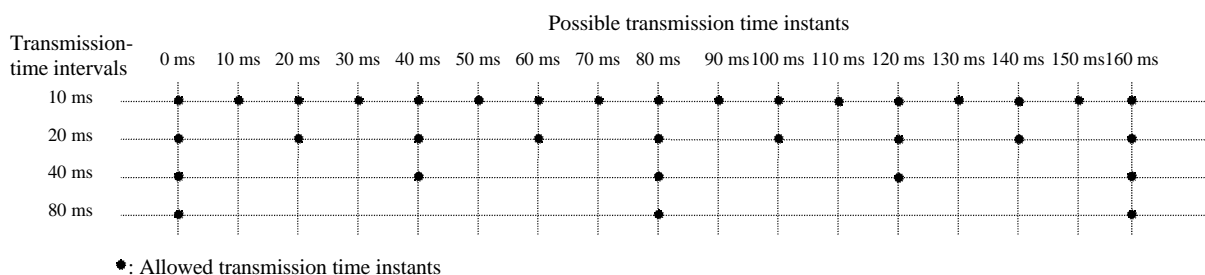
Bits on the PhCH  $P_t$  in timeslot  $t$  after physical channel mapping:

$$w_{P_t k} = v_{(k+U_1+\dots+U_{P_t-1})} \quad k = 1, 2, \dots, U_{P_t}$$

## 6.2.12 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 should have co-ordinated timings in the sense that transport blocks arriving from higher layers on different transport channels of potentially different transmission time intervals shall have aligned transmission time instants as shown in figure 6-6.
- 2) Different CCTrCHs cannot be mapped onto the same physical channel.
- 3) One CCTrCH shall be mapped onto one or several physical channels.



**Figure 6-5: Possible transmission time instants regarding CCTrCH**

- 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.

There may be one TFCI for each CCTrCH of dedicated type as well as for USCH- and DSCH-CCTrCHs.

### 6.2.12.1 Allowed CCTrCH combinations for one UE

#### 6.2.12.1.1 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

#### 6.2.12.1.2 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

## 6.2.13 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 is used, i.e. the receiver side uses the possible transport format combinations as a priori information.

### 6.2.13.1 Blind transport format detection

Blind transport format detection may be performed in the receiver by trying all possible combinations of the transport format.

## 6.2.13.2 Explicit transport format detection based on TFCI

### 6.2.13.2.1 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.

## 6.3 Coding for layer 1 control

### 6.3.1 Coding of transport format combination indicator (TFCI)

The number of TFCI bits is variable and is set at the beginning of the call via higher layer signalling. Encoding of the TFCI bits depends on the number of them. If there are 6-10 bits of TFCI the channel encoding is done as described in section 6.3.1.1. Also specific coding of less than 6 bits is possible as explained in 6.3.1.3. For improved TFCI detection reliability repetition is used to increase the number of TFCI bits. Additionally, with any TFCI coding scheme it is assumed that in the receiver combining of two successive TFCI words will be performed if the shortest transmission time interval of any TrCH is at least 20 ms.

#### 6.3.1.1 Default TFCI word

If the number of TFCI bits is 6-10 (32,10) sub-code of the second order Reed-Muller code is used. If the TFCI consist of 6-9bits, it is padded with zeros to 10 bits by setting the most significant bits to zero. The receiver can use the information that not all 10 bits are used for the TFCI, thereby reducing the error rate in the TFCI

TFCI is encoded by the (32,10) sub-code of second order Reed-Muller code. The code words of the (32,10) sub-code of second order Reed-Muller code are linear combination of some among 10 basis sequences: all 1's, 5 OVSF codes ( $C_{32,2}$ ,  $C_{32,3}$ ,  $C_{32,5}$ ,  $C_{32,9}$ ,  $C_{32,17}$ ), and 4 masks (Mask1, Mask2, Mask3, Mask4). The 4 mask sequences are as follows in Table 6.3.1-1.

**Table 6.3.1-1. Mask sequences**

|        |                                  |
|--------|----------------------------------|
| Mask 1 | 00101000011000111111000001110111 |
| Mask 2 | 00000001110011010110110111000111 |
| Mask 3 | 00001010111110010001101100101011 |
| Mask 4 | 00011100001101110010111101010001 |

For information bits  $a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9$  ( $a_0$  is LSB and  $a_9$  is MSB), the encoder structure is as follows in Figure 6-6.

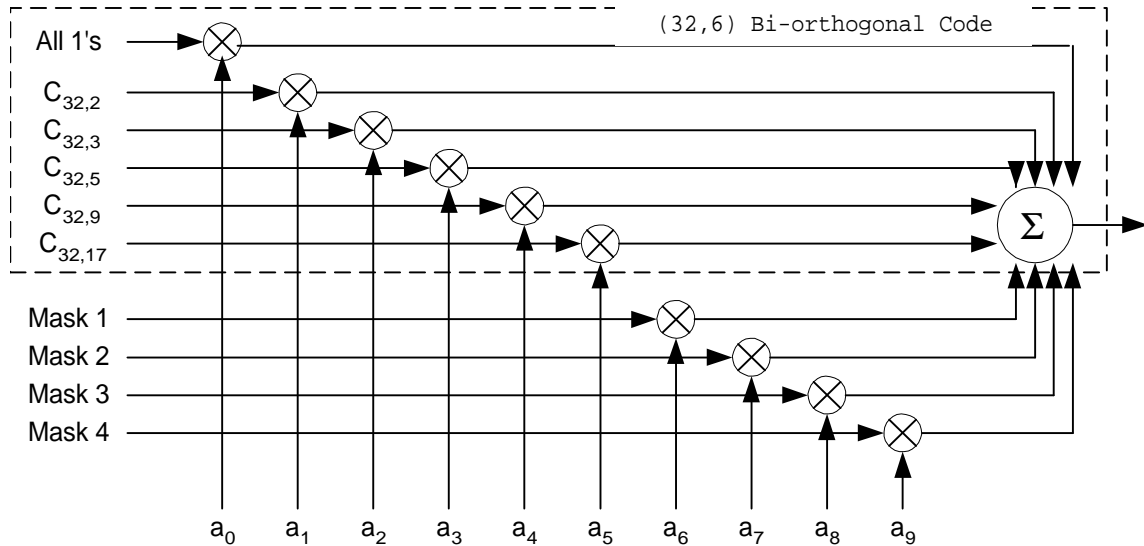


Figure 6-6. Encoder structure for (32,10) sub-code of second order Reed-Muller code

### 6.3.1.2 Coding of short TFCI lengths

If the number of TFCI bits is 1 or 2, then repetition will be used for coding. In this case each bit is repeated 3 times giving 4-bit transmission for a single TFCI bit and 8-bit transmission for 2 TFCI bits.

If the number of TFCI bits is in the range of 3 to 5, then one word of the biorthogonal (16,5) block code will be used.

The code words of the biorthogonal (16, 5) code are from two mutually biorthogonal sets,

$S_{C_4} = \{C_4(0), C_4(1), \dots, C_4(15)\}$  and its binary complement,  $\bar{S}_{C_4} = \{\bar{C}_4(0), \bar{C}_4(1), \dots, \bar{C}_4(15)\}$ . Words of set

$S_{C_4}$  are from the level 4 of the code three, which is generated, using the short code generation method defined in TS 25.223. The mapping of information bits to code words is shown in the Table 6.3.1-2.

Table 6.3.1- 2 Mapping of information bits to code words for biorthogonal (16, 5) code.

| Information bits | Code word       |
|------------------|-----------------|
| 00000            | $C_4(0)$        |
| 00001            | $\bar{C}_4(0)$  |
| 00010            | $C_4(1)$        |
| ...              | ...             |
| 11101            | $\bar{C}_4(14)$ |
| 11110            | $C_4(15)$       |
| 11111            | $\bar{C}_4(15)$ |

### 6.3.2 Operation of Transport-format-combination indicator (TFCI) in Split Mode

In the case of DCH in Split Mode, the UTRAN shall operate with as follows:

- If one of the links is associated with a DSCH, the TFCI code word may be split into two code words of length 16. The use of such a functionality shall be indicated by higher layer signalling.

TFCI information is encoded by biorthogonal (16, 5) block code. The code words of the biorthogonal (16, 5) code are from two mutually biorthogonal sets,  $S_{C_{16}} = \{C_{16,1}, C_{16,2}, \dots, C_{16,16}\}$  and its binary complement,  $\bar{S}_{C_{16}} = \{\bar{C}_{16,1}, \bar{C}_{16,2}, \dots, \bar{C}_{16,16}\}$ . Code words of set  $S_{C_{16}}$  are from the level 16 of the code three of OVSF codes defined in document TS 25.213. The mapping of information bits to code words is shown in the Table 2.

**Table 6.3.2-1: Mapping of information bits to code words for biorthogonal (16, 5) code**

| Information bits | Code word         |
|------------------|-------------------|
| 00000            | $C_{16,1}$        |
| 00001            | $\bar{C}_{16,1}$  |
| 00010            | $C_{16,2}$        |
| ...              | ...               |
| 11101            | $\bar{C}_{16,15}$ |
| 11110            | $C_{16,16}$       |
| 11111            | $\bar{C}_{16,16}$ |

### 6.3.3 Coding of Paging Indicator (PI)

The PI is an identifier to instruct the UE whether there is a paging message for the groups of mobiles that are associated to the PI. The length  $L_{PI}$  of the PI is  $L_{PI}=2$ ,  $L_{PI}=4$  or  $L_{PI}=8$  symbols. The coding of the PI is shown in table 6.3.3-1.

**Table 6.3.3-1 Coding of the PI**

| Bits    | PI      | Content                              |
|---------|---------|--------------------------------------|
| All '0' | not set | There is no necessity to receive PCH |
| All '1' | set     | There is necessity to receive PCH-   |

### 6.3.4 Coding of Transmit Power Control (TPC)

The TPC command is an identifier sent in uplink transmission only, to instruct the NodeB whether Tx power has to be increased or decreased. The length of the TPC command is one symbol. The coding of the TPC command is shown in table 6.3.3-1.

**Table 6.3.4-1 Coding of the TPC**

| TPC    | TPC Bits | Meaning           |
|--------|----------|-------------------|
| 'Down' | 00       | Decrease Tx Power |
| 'Up'   | 11       | Increase Tx Power |



## 7 History

| <b>Document history</b> |            |   |
|-------------------------|------------|---|
| v0.0.1                  | 1999-02-12 | Document created based on the documents Tdoc UMTS (xx.10) V0.5.0 and ARIB Volume 3 Version 1.0-1.0. |
| V0.0.2                  | 1999-02-18 | Document updated based on comments in the TSG-RAN WG1 e-mail reflector.                             |
| V0.1.0                  | 1999-02-26 | This version was approved at WG1#2 as the initial merge of ARIB and ETSI documents.                 |
| V0.1.1                  | 1999-03-03 | Document updated based on Adhoc decisions and technical comments made at WG1#2.                     |
| V1.0.1                  | 1999-03-23 | Document updated based on AdHoc decisions made at WG1#3.  |
| V1.0.2                  | 1999-03-24 | Corrections made based on feedback in WG1#3 meeting.  |
| V1.1.0                  | 1999-03-26 | This version was approved at WG1#3 meeting.   |
| V1.1.1                  | 1999-04-19 | Document updated based on AdHoc decisions at WG1#4.   |
| V1.1.2                  | 1999-04-20 | Document updated based on additional text proposals at WG1#4.                                       |
| V2.0.0                  | 1999-04-20 | This version was approved at WG1#4 as an input to TSG RAN meeting.                                  |
| TS 25.222 V1.0.0        | 1999-04-22 | Noted by TSG-RAN as TS 25.222 V1.0.0  |
| TS 25.222 V1.0.1        | 1999-06-04 | Updated based on AdHoc decisions at WG1#5   |
| TS 25.222 V1.1.0        | 1999-06-04 | This version was approved at WG1#5 meeting.   |
| TS 25.222 V2.0.0        | 1999-06-22 | Endorsed by TSG-RAN as V2.0.0   |
| TS 25.222 V2.0.1        | 1999-08-03 | Updated based on harmonization decisions and Ad Hoc decisions at WG1#6                              |
| TS 25.222 V2.0.2        | 1999-08-23 | Editorial corrections based on feedback on WG1 reflector.   |
| TS 25.222 V2.1.0        | 1999-08-30 | Accepted as an output of WG1#6 meeting at WG1#7   |

|  |            |   |
|--|------------|---|
| TS 25.222 V2.1.1   | 1999-09-02 | Document updated based on text proposals at WG1#7 |
| <p>The editor for TS 25.222 Multiplexing and channel coding (TDD), is:</p> <p>Jussi Kahtava<br/>Nokia-Japan<br/>Email: jussi.kahtava@nokia.com</p> <p>This document is written in Microsoft Word 97.</p> |            |   |