



**E**UROPEAN  
**T**ELECOMMUNICATION  
**S**TANDARD

**ETS 300 900**

April 1997

---

Source: ETSI TC-SMG

Reference: DE/SMG-040338QR3

ICS: 33.020

**Key words:** Digital cellular telecommunications system, Global System for Mobile communications (GSM)



**Digital cellular telecommunications system (Phase 2+);  
Alphabets and language-specific information  
(GSM 03.38 version 5.4.1)**

**ETSI**

European Telecommunications Standards Institute

**ETSI Secretariat**

**Postal address:** F-06921 Sophia Antipolis CEDEX - FRANCE

**Office address:** 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE

**X.400:** c=fr, a=atlas, p=etsi, s=secretariat - **Internet:** secretariat@etsi.fr

Tel.: +33 4 92 94 42 00 - Fax: +33 4 93 65 47 16

---

**Copyright Notification:** No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 1997. All rights reserved.



## Contents

|   |    |
|---|----|
| Foreword .....                              | 5  |
| 1 Scope .....                               | 7  |
| 2 Normative references .....                | 7  |
| 3 Abbreviations.....                        | 7  |
| 4 SMS Data Coding Scheme .....              | 8  |
| 5 Cell Broadcast Data Coding Scheme.....    | 11 |
| 6 Individual parameters .....               | 13 |
| 6.1 General principles .....                | 13 |
| 6.1.1 General notes.....                    | 13 |
| 6.1.2 Character packing .....               | 13 |
| 6.1.2.1 SMS Point-to-Point Packing .....    | 13 |
| 6.1.2.1.1 Packing of 7-bit characters ..... | 13 |
| 6.1.2.2 SMS Cell Broadcast Packing.....     | 14 |
| 6.1.2.2.1 Packing of 7-bit characters ..... | 14 |
| 6.1.2.3 USSD packing .....                  | 15 |
| 6.1.2.3.1 Packing of 7 bit characters ..... | 15 |
| 6.2 Alphabet tables .....                   | 18 |
| 6.2.1 Default alphabet .....                | 18 |
| History.....                                | 19 |

Blank page

## Foreword

This European Telecommunication Standard (ETS) has been produced by the Special Mobile Group (SMG) Technical Committee (TC) of the European Telecommunications Standards Institute (ETSI).

This ETS defines the language-specific requirements for GSM within the digital cellular telecommunications system (Phase 2+).

This ETS is a GSM technical specification version 5, which incorporates GSM Phase 2+ enhancements/features to the version 4 GSM technical specification. The ETS from which this Phase 2+ ETS has evolved is Phase 2 GSM ETS 300 628 (GSM 03.38 version 4.0.1).

The contents of this ETS is subject to continuing work within TC-SMG and may change following formal TC-SMG approval. Should TC-SMG modify the contents of this ETS, it will be resubmitted for OAP by ETSI with an identifying change of release date and an increase in version number as follows:

Version 5.x.y

where:

- y the third digit is incremented when editorial only changes have been incorporated in the specification;
- x the second digit is incremented for all other types of changes, i.e. technical enhancements, corrections, updates, etc.

| <b>Transposition dates</b>  |                 |
|---|-----------------|
| Date of adoption:   | 21 March 1997   |
| Date of latest announcement of this ETS (doa):  | 31 July 1997    |
| Date of latest publication of new National Standard or endorsement of this ETS (dop/e): | 31 January 1998 |
| Date of withdrawal of any conflicting National Standard (dow):                          | 31 January 1998 |

Blank page

## 1 Scope

This European Telecommunication Standard (ETS) defines the language-specific requirements for GSM. These are specific codepoints required by the Short Message Service (SMS) specifications which in turn are used not only for SMS (GSM 03.40, 03.41) but also for Unstructured Data (GSM 02.90) and may additionally be used for Man-Machine Interface (MMI) (GSM 02.30).

The specification for the Data Circuit terminating Equipment/Data Terminal Equipment (DCE/DTE) interface (GSM 07.05 [8]) will also use the codes specified herein for the transfer of SMS data to an external terminal.

## 2 Normative references

This ETS incorporates by dated and undated reference, provisions from other publications. These normative references are cited at the appropriate places in the text and the publications are listed hereafter. For dated references, subsequent amendments to or revisions of any of these publications apply to this ETS only when incorporated in it by amendment or revision. For undated references, the latest edition of the publication referred to applies.

- [1] GSM 01.04 (ETR 350): "Digital cellular telecommunications system (Phase 2+); Abbreviations and acronyms".
- [2] GSM 02.30 (ETS 300 907): "Digital cellular telecommunications system (Phase 2+); Man-Machine Interface (MMI) of the Mobile Station (MS)".
- [3] GSM 03.90: "Digital cellular telecommunications system; Unstructured Supplementary Services Data (USSD) - Stage 2".
- [4] GSM 03.40 (ETS 300 901): "Digital cellular telecommunications system (Phase 2+); Technical realization of the Short Message Service (SMS) Point-to-Point (PP)".
- [5] GSM 03.41 (ETS 300 902): "Digital cellular telecommunications system (Phase 2+); Technical realization of Short Message Service Cell Broadcast (SMSCB)".
- [6] GSM 04.11 (ETS 300 942): "Digital cellular telecommunications system (Phase 2+); Point-to-Point (PP) Short Message Service (SMS) support on mobile radio interface".
- [7] GSM 04.12 (ETS 300 943): "Digital cellular telecommunications system (Phase 2+); Short Message Service Cell Broadcast (SMSCB) support on the mobile radio interface".
- [8] GSM 07.05: "Digital cellular telecommunications system (Phase 2+); Use of Data Terminal Equipment - Data Circuit terminating Equipment (DTE - DCE) interface for Short Message Service (SMS) and Cell Broadcast Service (CBS)".
- [10] ISO/IEC10646: "Universal Multiple-Octet Coded Character Set (UCS); UCS2, 16 bit coding".
- [11] GSM 04.90 (ETS 300 957): "Digital cellular telecommunications system; Unstructured Supplementary Services Data (USSD) - Stage 3".
- [12] ISO 639: "Code for the representation of names of languages".

## 3 Abbreviations

Abbreviations used in this ETS are listed in GSM 01.04.

## 4 SMS Data Coding Scheme

The TP-Data-Coding-Scheme field, defined in GSM 03.40, indicates the data coding scheme of the TP-UD field, and may indicate a message class. Any reserved codings shall be assumed to be the GSM default alphabet (the same as codepoint 00000000) by a receiving entity. The octet is used according to a coding group which is indicated in bits 7..4. The octet is then coded as follows:

| Coding Group Bits<br>7..4 | Use of bits 3..0  |  |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
|---------------------------|---|--|-------|---------------|---|---|---------|---|---|---------------------------------------|---|---|------------------------------|---|---|--|-------|------|-----------|---|---|------------------|---|---|-------|---|---|-------------------|---|---|----------|
| 00xx                      | <p>General Data Coding indication<br/>Bits 5..0 indicate the following:</p> <p>Bit 5, if set to 0, indicates the text is uncompressed<br/>Bit 5, if set to 1, indicates the text is compressed using the GSM standard compression algorithm. (yet to be specified)</p> <p>Bit 4, if set to 0, indicates that bits 1 to 0 are reserved and have no message class meaning<br/>Bit 4, if set to 1, indicates that bits 1 to 0 have a message class meaning:</p> <table border="0"> <tr> <td>Bit 1</td> <td>Bit 0</td> <td>Message Class</td> </tr> <tr> <td>0</td> <td>0</td> <td>Class 0</td> </tr> <tr> <td>0</td> <td>1</td> <td>Class 1 Default meaning: ME-specific.</td> </tr> <tr> <td>1</td> <td>0</td> <td>Class 2 SIM specific message</td> </tr> <tr> <td>1</td> <td>1</td> <td>Class 3 Default meaning: TE specific (see GSM 07.05 [8])</td> </tr> </table> <p>Bits 3 and 2 indicate the alphabet being used, as follows:</p> <table border="0"> <tr> <td>Bit 3</td> <td>Bit2</td> <td>Alphabet:</td> </tr> <tr> <td>0</td> <td>0</td> <td>Default alphabet</td> </tr> <tr> <td>0</td> <td>1</td> <td>8 bit</td> </tr> <tr> <td>1</td> <td>0</td> <td>UCS2 (16bit) [10]</td> </tr> <tr> <td>1</td> <td>1</td> <td>Reserved</td> </tr> </table> <p>NOTE: The special case of bits 7..0 being 0000 0000 indicates the Default Alphabet as in Phase 2</p> | Bit 1  | Bit 0 | Message Class | 0 | 0 | Class 0 | 0 | 1 | Class 1 Default meaning: ME-specific. | 1 | 0 | Class 2 SIM specific message | 1 | 1 | Class 3 Default meaning: TE specific (see GSM 07.05 [8]) | Bit 3 | Bit2 | Alphabet: | 0 | 0 | Default alphabet | 0 | 1 | 8 bit | 1 | 0 | UCS2 (16bit) [10] | 1 | 1 | Reserved |
| Bit 1                     | Bit 0   | Message Class  |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| 0                         | 0   | Class 0  |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| 0                         | 1   | Class 1 Default meaning: ME-specific.                    |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| 1                         | 0   | Class 2 SIM specific message                             |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| 1                         | 1   | Class 3 Default meaning: TE specific (see GSM 07.05 [8]) |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| Bit 3                     | Bit2  | Alphabet:  |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| 0                         | 0   | Default alphabet   |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| 0                         | 1   | 8 bit  |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| 1                         | 0   | UCS2 (16bit) [10]  |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| 1                         | 1   | Reserved   |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| 0100..1011                | Reserved coding groups  |  |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |
| 1100                      | <p>Message Waiting Indication Group: Discard Message</p> <p>Bits 3..0 are coded exactly the same as Group 1101, however with bits 7..4 set to 1100 the mobile may discard the contents of the message, and only present the indication to the user.</p>   |  |       |               |   |   |         |   |   |                                       |   |   |                              |   |   |  |       |      |           |   |   |                  |   |   |       |   |   |                   |   |   |          |

(continued)



(concluded)

|      |   |  |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
|------|---|--|-------------------------|---|-----------------------|---|---|---------------------------|---|---|--------------------------------------|---|---|---------------------------------|---|---|--|
| 1101 | <p>Message Waiting Indication Group: Store Message</p> <p>This Group allows an indication to be provided to the user about the status of types of message waiting on systems connected to the GSM PLMN. The mobile may present this indication as an icon on the screen, or other MMI indication. The mobile may take note of the Origination Address for messages in this group and group 1100. For each indication supported, the mobile may provide storage for the Origination Address which is to control the mobile indicator. Text included in the user data is coded in the Default Alphabet. Where a message is received with bits 7..4 set to 1101, the mobile shall store the text of the SMS message in addition to setting the indication.</p> <p>Bits 3 indicates Indication Sense:</p> <p>Bit 3</p> <table border="0"> <tr><td>0</td><td>Set Indication Inactive</td></tr> <tr><td>1</td><td>Set Indication Active</td></tr> </table> <p>Bit 2 is reserved, and set to 0</p> <p>Bit 1 Bit 0 Indication Type:</p> <table border="0"> <tr><td>0</td><td>0</td><td>Voicemail Message Waiting</td></tr> <tr><td>0</td><td>1</td><td>Fax Message Waiting</td></tr> <tr><td>1</td><td>0</td><td>Electronic Mail Message Waiting</td></tr> <tr><td>1</td><td>1</td><td>Other Message Waiting*</td></tr> </table> <p>* Mobile manufacturers may implement the "Other Message Waiting" indication as an additional indication without specifying the meaning. The meaning of this indication is intended to be standardized in the future, so Operators should not make use of this indication until the standard for this indication is finalized.</p> | 0  | Set Indication Inactive | 1 | Set Indication Active | 0 | 0 | Voicemail Message Waiting | 0 | 1 | Fax Message Waiting                  | 1 | 0 | Electronic Mail Message Waiting | 1 | 1 | Other Message Waiting*                                   |
| 0    | Set Indication Inactive   |  |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 1    | Set Indication Active   |  |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 0    | 0   | Voicemail Message Waiting                                |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 0    | 1   | Fax Message Waiting                                      |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 1    | 0   | Electronic Mail Message Waiting                          |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 1    | 1   | Other Message Waiting*                                   |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 1110 | <p>Message Waiting Indication Group: Store Message</p> <p>The coding of bits 3..0 and functionality of this feature are the same as for the Message Waiting Indication Group above, (bits 7..4 set to 1101) with the exception that the text included in the user data is coded in the uncompressed UCS2 alphabet.</p>  |  |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 1111 | <p>Data coding/message class</p> <p>Bit 3 is reserved, set to 0.</p> <p>Bit 2 Message coding:</p> <table border="0"> <tr><td>0</td><td>Default alphabet</td></tr> <tr><td>1</td><td>8-bit data</td></tr> </table> <p>Bit 1 Bit 0 Message Class:</p> <table border="0"> <tr><td>0</td><td>0</td><td>Class 0</td></tr> <tr><td>0</td><td>1</td><td>Class 1 default meaning: ME-specific</td></tr> <tr><td>1</td><td>0</td><td>Class 2 SIM-specific message</td></tr> <tr><td>1</td><td>1</td><td>Class 3 default meaning: TE specific (see GSM 07.05 [8])</td></tr> </table>  | 0  | Default alphabet        | 1 | 8-bit data            | 0 | 0 | Class 0                   | 0 | 1 | Class 1 default meaning: ME-specific | 1 | 0 | Class 2 SIM-specific message    | 1 | 1 | Class 3 default meaning: TE specific (see GSM 07.05 [8]) |
| 0    | Default alphabet  |  |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 1    | 8-bit data  |  |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 0    | 0   | Class 0  |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 0    | 1   | Class 1 default meaning: ME-specific                     |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 1    | 0   | Class 2 SIM-specific message                             |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |
| 1    | 1   | Class 3 default meaning: TE specific (see GSM 07.05 [8]) |                         |   |                       |   |   |                           |   |   |                                      |   |   |                                 |   |   |  |

Default alphabet indicates that the TP-UD is coded from the 7-bit alphabet given in subclause 6.2.1. When this alphabet is used, the characters of the message are packed in octets as shown in subclause 6.1.2.1.1, and the message can consist of up to 160 characters. The default alphabet shall be supported by all MSs and SCs offering the service.

8-bit data indicates that the TP-UD has user-defined coding, and the message can consist of up to 140 octets.

UCS2 alphabet indicates that the TP-UD has a UCS2 [10] coded message, and the message can consist of up to 140 octets, i.e. up to 70 UCS2 characters. The General notes specified in subclause 6.1.1 override any contrary specification in UCS2, so for example even in UCS2 a <CR> character will cause

the MS to return to the beginning of the current line and overwrite any existing text with the characters which follow the <CR>.

When a message is compressed, the TP-UD consists of the default alphabet or UCS2 alphabet compressed message, and the compressed message itself can consist of up to 140 octets in total.

When a mobile terminated message is class 0 and the MS has the capability of displaying short messages, the MS shall display the message immediately and send an acknowledgement to the SC when the message has successfully reached the MS irrespective of whether there is memory available in the SIM or ME. The message shall not be automatically stored in the SIM or ME.

The ME may make provision through MMI for the user to selectively prevent the message from being displayed immediately.

If the ME is incapable of displaying short messages or if the immediate display of the message has been disabled through MMI then the ME shall treat the short message as though there was no message class, i.e. it will ignore bits 0 and 1 in the TP-DCS and normal rules for memory capacity exceeded shall apply.

When a mobile terminated message is Class 1, the MS shall send an acknowledgement to the SC when the message has successfully reached the MS and can be stored. The MS shall normally store the message in the ME by default, if that is possible, but otherwise the message may be stored elsewhere, e.g. in the SIM. The user may be able to override the default meaning and select their own routing.

When a mobile terminated message is Class 2 (SIM-specific), a phase 2 (or later) MS shall ensure that the message has been transferred to the SMS data field in the SIM before sending an acknowledgement to the SC. The MS shall return a "protocol error, unspecified" error message (see GSM 04.11) if the short message cannot be stored in the SIM and there is other short message storage available at the MS. If all the short message storage at the MS is already in use, the MS shall return "memory capacity exceeded".

When a mobile terminated message is Class 3, the MS shall send an acknowledgement to the SC when the message has successfully reached the MS and can be stored, irrespectively of whether the MS supports an SMS interface to a TE, and without waiting for the message to be transferred to the TE. Thus the acknowledgement to the SC of a TE-specific message does not imply that the message has reached the TE. Class 3 messages shall normally be transferred to the TE when the TE requests "TE-specific" messages (see GSM 07.05 [8]). The user may be able to override the default meaning and select their own routing.

The message class codes may also be used for mobile originated messages, to provide an indication to the destination SME of how the message was handled at the MS.

The MS will not interpret reserved or unsupported values but shall store them as received. The SC may reject messages with a Data Coding Scheme containing a reserved value or one which is not supported.

## 5 Cell Broadcast Data Coding Scheme

The Cell Broadcast Data Coding Scheme indicates the intended handling of the message at the MS, the alphabet/coding, and the language (when applicable). Any reserved codings shall be assumed to be the GSM default alphabet (the same as codepoint 00001111) by a receiving entity. The octet is used according to a coding group which is indicated in bits 7..4. The octet is then coded as follows:

| Coding Group<br>Bits<br>7..4 | Use of bits 3..0   |
|------------------------------|--|
| 0000                         | Language using the default alphabet<br><br>Bits 3..0 indicate the language:<br>0000 German<br>0001 English<br>0010 Italian<br>0011 French<br>0100 Spanish<br>0101 Dutch<br>0110 Swedish<br>0111 Danish<br>1000 Portuguese<br>1001 Finnish<br>1010 Norwegian<br>1011 Greek<br>1100 Turkish<br>1101..1110 Reserved for European languages<br>1111 Language unspecified   |
| 0001                         | 0000 Default alphabet; message preceded by language indication<br><br>The first 3 characters of the message are a two-character representation of the language encoded according to ISO 639 [12], followed by a CR character. The CR character is then followed by 90 characters of text. Older mobiles will overwrite the start of the message up to the CR and present only the text.<br><br>0001 UCS2; message preceded by language indication<br><br>The message starts with a two 7-bit default alphabet character representation of the language encoded according to ISO 639 [12]. This is padded to the octet boundary with two bits set to 0 and then followed by 40 characters of UCS2-encoded message. Older mobiles will display the two-character language identifier and then an apparently-random string of characters.<br><br>0010..1111 Reserved for European languages |
| 0010..0011                   | Reserved for European Languages using the default alphabet, with unspecified handling at the MS  |

(continued)

(concluded)

|                              |  |  |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
|------------------------------|--|--|-----------------|----------------|------------------|---|-----------|-------|-------|---------------------------------------|---|---|-------------------------------|---|---|--|-------|-------|----------------------|---|---|------------------|---|---|-------|---|---|--------------------|---|---|----------|
| 01xx                         | <p>General Data Coding indication<br/>                 Bits 5..0 indicate the following:</p> <p>Bit 5, if set to 0, indicates the text is uncompressed<br/>                 Bit 5, if set to 1, indicates the text is compressed using the GSM standard compressing algorithm. (yet to be specified)</p> <p>Bit 4, if set to 0, indicates that bits 1 to 0 are reserved and have no message class meaning<br/>                 Bit 4, if set to 1, indicates that bits 1 to 0 have a message class meaning:</p> <table border="0"> <tr> <td>Bit 1</td> <td>Bit 0</td> <td>Message Class:</td> </tr> <tr> <td>0</td> <td>0</td> <td>Class 0</td> </tr> <tr> <td>0</td> <td>1</td> <td>Class 1 Default meaning: ME-specific.</td> </tr> <tr> <td>1</td> <td>0</td> <td>Class 2 SIM specific message.</td> </tr> <tr> <td>1</td> <td>1</td> <td>Class 3 Default meaning: TE-specific (see GSM 07.05 [8])</td> </tr> </table> <p>Bits 3 and 2 indicate the alphabet being used, as follows:</p> <table border="0"> <tr> <td>Bit 3</td> <td>Bit 2</td> <td>Alphabet:</td> </tr> <tr> <td>0</td> <td>0</td> <td>Default alphabet</td> </tr> <tr> <td>0</td> <td>1</td> <td>8 bit</td> </tr> <tr> <td>1</td> <td>0</td> <td>USC2 (16 bit) [10]</td> </tr> <tr> <td>1</td> <td>1</td> <td>Reserved</td> </tr> </table> | Bit 1  | Bit 0           | Message Class: | 0                | 0 | Class 0   | 0     | 1     | Class 1 Default meaning: ME-specific. | 1 | 0 | Class 2 SIM specific message. | 1 | 1 | Class 3 Default meaning: TE-specific (see GSM 07.05 [8]) | Bit 3 | Bit 2 | Alphabet:            | 0 | 0 | Default alphabet | 0 | 1 | 8 bit | 1 | 0 | USC2 (16 bit) [10] | 1 | 1 | Reserved |
| Bit 1                        | Bit 0  | Message Class:   |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 0                            | 0  | Class 0  |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 0                            | 1  | Class 1 Default meaning: ME-specific.                    |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 1                            | 0  | Class 2 SIM specific message.                            |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 1                            | 1  | Class 3 Default meaning: TE-specific (see GSM 07.05 [8]) |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| Bit 3                        | Bit 2  | Alphabet:  |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 0                            | 0  | Default alphabet   |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 0                            | 1  | 8 bit  |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 1                            | 0  | USC2 (16 bit) [10]                                       |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 1                            | 1  | Reserved   |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| Coding Group<br>Bits<br>7..4 | Use of bits 3..0   |  |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 1000..1110                   | Reserved coding groups   |  |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 1111                         | <p>Data coding / message handling</p> <p>Bit 3 is reserved, set to 0.</p> <table border="0"> <tr> <td>Bit 2</td> <td>Message coding:</td> </tr> <tr> <td>0</td> <td>Default alphabet</td> </tr> <tr> <td>1</td> <td>8bit data</td> </tr> </table> <table border="0"> <tr> <td>Bit 1</td> <td>Bit 0</td> <td>Message Class:</td> </tr> <tr> <td>0</td> <td>0</td> <td>No message class</td> </tr> <tr> <td>0</td> <td>1</td> <td>Class 1 user defined</td> </tr> <tr> <td>1</td> <td>0</td> <td>Class 2 user defined</td> </tr> <tr> <td>1</td> <td>1</td> <td>Class 3</td> </tr> </table> <p>default meaning: TE specific<br/>                 (see GSM 07.05 [8])</p>   | Bit 2  | Message coding: | 0              | Default alphabet | 1 | 8bit data | Bit 1 | Bit 0 | Message Class:                        | 0 | 0 | No message class              | 0 | 1 | Class 1 user defined                                     | 1     | 0     | Class 2 user defined | 1 | 1 | Class 3          |   |   |       |   |   |                    |   |   |          |
| Bit 2                        | Message coding:  |  |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 0                            | Default alphabet   |  |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 1                            | 8bit data  |  |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| Bit 1                        | Bit 0  | Message Class:   |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 0                            | 0  | No message class   |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 0                            | 1  | Class 1 user defined                                     |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 1                            | 0  | Class 2 user defined                                     |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |
| 1                            | 1  | Class 3  |                 |                |                  |   |           |       |       |                                       |   |   |                               |   |   |  |       |       |                      |   |   |                  |   |   |       |   |   |                    |   |   |          |

These codings may also be used for Unstructured SS Data and MMI/display purposes.

See GSM 04.90 [11] for specific coding values applicable to Unstructured SS Data for MS originated USSD messages and MS terminated USSD messages. USSD messages using the default alphabet are coded with the 7-bit alphabet given in subclause 6.2.1. The message can then consist of up to 182 user characters.

Cell Broadcast messages using the default alphabet are coded with the 7-bit alphabet given in subclause 6.2.1. The message then consists of 93 user characters.

Cell Broadcast messages using 8-bit data have user-defined coding, and will be 82 octets in length.

UCS2 alphabet indicates that the message is coded in UCS2 [10]. The General notes specified in subclause 6.1.1 override any contrary specification in UCS2, so for example even in UCS2 a <CR> character will cause the MS to return to the beginning of the current line and overwrite any existing text with the characters which follow the <CR>. Messages encoded in UCS2 consist of 41 characters.

Class 1 and Class 2 messages may be routed by the ME to user-defined destinations, but the user may override any default meaning and select their own routing.

Class 3 messages will normally be selected for transfer to a TE, in cases where a ME supports an SMS/CBS interface to a TE, and the TE requests "TE-specific" cell broadcast messages (see GSM 07.05 [8]). The user may be able to override the default meaning and select their own routing.

## 6 Individual parameters

### 6.1 General principles

#### 6.1.1 General notes

Except where otherwise indicated, the following shall apply to all alphabet tables:

- 1: The characters marked "1)" are not used but are displayed as a space.
- 2: The characters of this set, when displayed, should approximate to the appearance of the relevant characters specified in ISO 1073 and the relevant national standards.
- 3: Control characters:
 

| Code | Meaning.  |
|------|---|
| LF   | Line feed: Any characters following LF which are to be displayed shall be presented as the next line of the message, commencing with the first character position.          |
| CR   | Carriage return: Any characters following CR which are to be displayed shall be presented as the current line of the message, commencing with the first character position. |
| SP   | Space character.  |
- 4: The display of characters within a message is achieved by taking each character in turn and placing it in the next available space from left to right and top to bottom.

#### 6.1.2 Character packing

##### 6.1.2.1 SMS Point-to-Point Packing

###### 6.1.2.1.1 Packing of 7-bit characters

If a character number  $\alpha$  is noted in the following way:

|            |            |            |            |            |            |            |
|------------|------------|------------|------------|------------|------------|------------|
| b7         | b6         | b5         | b4         | b3         | b2         | b1         |
| $\alpha a$ | $\alpha b$ | $\alpha c$ | $\alpha d$ | $\alpha e$ | $\alpha f$ | $\alpha g$ |

The packing of the 7-bits characters in octets is done by completing the octets with zeros on the left.

For examples, packing:  $\alpha$

- one character in one octet:

- bits number:

|   |    |    |    |    |    |    |    |
|---|----|----|----|----|----|----|----|
| 7 | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
| 0 | 1a | 1b | 1c | 1d | 1e | 1f | 1g |

- two characters in two octets:

- bits number:

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
| 2g | 1a | 1b | 1c | 1d | 1e | 1f | 1g |
| 0  | 0  | 2a | 2b | 2c | 2d | 2e | 2f |

- three characters in three octets:

- bits number:

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
| 2g | 1a | 1b | 1c | 1d | 1e | 1f | 1g |
| 3f | 3g | 2a | 2b | 2c | 2d | 2e | 2f |
| 0  | 0  | 0  | 3a | 3b | 3c | 3d | 3e |

- seven characters in seven octets:

- bits number:

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
| 2g | 1a | 1b | 1c | 1d | 1e | 1f | 1g |
| 3f | 3g | 2a | 2b | 2c | 2d | 2e | 2f |
| 4e | 4f | 4g | 3a | 3b | 3c | 3d | 3e |
| 5d | 5e | 5f | 5g | 4a | 4b | 4c | 4d |
| 6c | 6d | 6e | 6f | 6g | 5a | 5b | 5c |
| 7b | 7c | 7d | 7e | 7f | 7g | 6a | 6b |
| 0  | 0  | 0  | 0  | 0  | 0  | 0  | 7a |

- eight characters in seven octets:

- bits number:

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
| 2g | 1a | 1b | 1c | 1d | 1e | 1f | 1g |
| 3f | 3g | 2a | 2b | 2c | 2d | 2e | 2f |
| 4e | 4f | 4g | 3a | 3b | 3c | 3d | 3e |
| 5d | 5e | 5f | 5g | 4a | 4b | 4c | 4d |
| 6c | 6d | 6e | 6f | 6g | 5a | 5b | 5c |
| 7b | 7c | 7d | 7e | 7f | 7g | 6a | 6b |
| 8a | 8b | 8c | 8d | 8e | 8f | 8g | 7a |

The bit number zero is always transmitted first.

Therefore, in 140 octets, it is possible to pack  $(140 \times 8) / 7 = 160$  characters.

### 6.1.2.2 SMS Cell Broadcast Packing

#### 6.1.2.2.1 Packing of 7-bit characters

If a character number  $\alpha$  is noted in the following way:

|            |            |            |            |            |            |            |
|------------|------------|------------|------------|------------|------------|------------|
| b7         | b6         | b5         | b4         | b3         | b2         | b1         |
| $\alpha a$ | $\alpha b$ | $\alpha c$ | $\alpha d$ | $\alpha e$ | $\alpha f$ | $\alpha g$ |

the packing of the 7-bits characters in octets is done as follows:

| bit number   |  | 7   | 6   | 5   | 4   | 3   | 2   | 1   | 0   |
|--------------|--|-----|-----|-----|-----|-----|-----|-----|-----|
| octet number |  |     |     |     |     |     |     |     |     |
| 1            |  | 2g  | 1a  | 1b  | 1c  | 1d  | 1e  | 1f  | 1g  |
| 2            |  | 3f  | 3g  | 2a  | 2b  | 2c  | 2d  | 2e  | 2f  |
| 3            |  | 4e  | 4f  | 4g  | 3a  | 3b  | 3c  | 3d  | 3e  |
| 4            |  | 5d  | 5e  | 5f  | 5g  | 4a  | 4b  | 4c  | 4d  |
| 5            |  | 6c  | 6d  | 6e  | 6f  | 6g  | 5a  | 5b  | 5c  |
| 6            |  | 7b  | 7c  | 7d  | 7e  | 7f  | 7g  | 6a  | 6b  |
| 7            |  | 8a  | 8b  | 8c  | 8d  | 8e  | 8f  | 8g  | 7a  |
| 8            |  | 10g | 9a  | 9b  | 9c  | 9d  | 9e  | 9f  | 9g  |
|              |  |     |     |     |     |     |     |     |     |
|              |  |     |     |     |     |     |     |     |     |
|              |  |     |     |     |     |     |     |     |     |
|              |  |     |     |     |     |     |     |     |     |
|              |  |     |     |     |     |     |     |     |     |
|              |  |     |     |     |     |     |     |     |     |
|              |  |     |     |     |     |     |     |     |     |
| 81           |  | 93d | 93e | 93f | 93g | 92a | 92b | 92c | 92d |
| 82           |  | 0   | 0   | 0   | 0   | 0   | 93a | 93b | 93c |

The bit number zero is always transmitted first.

Therefore, in 82 octets, it is possible to pack  $(82 \times 8) / 7 = 93.7$ , that is 93 characters. The 5 remaining bits are set to zero as stated above.

### 6.1.2.3 USSD packing

#### 6.1.2.3.1 Packing of 7 bit characters

If a character number  $\alpha$  is noted in the following way:

| b7         | b6         | b5         | b4         | b3         | b2         | b1         |
|------------|------------|------------|------------|------------|------------|------------|
| $\alpha a$ | $\alpha b$ | $\alpha c$ | $\alpha d$ | $\alpha e$ | $\alpha f$ | $\alpha g$ |

The packing of the 7-bit characters in octets is done by completing the octets with zeros on the left.

For example, packing:  $\alpha$

- one character in one octet:

- bits number:

| 7 | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|---|----|----|----|----|----|----|----|
| 0 | 1a | 1b | 1c | 1d | 1e | 1f | 1g |

- two characters in two octets:

- bits number:

| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|----|----|----|----|----|----|----|----|
| 2g | 1a | 1b | 1c | 1d | 1e | 1f | 1g |
| 0  | 0  | 2a | 2b | 2c | 2d | 2e | 2f |

- three characters in three octets:

- bits number:

| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|----|----|----|----|----|----|----|----|
| 2g | 1a | 1b | 1c | 1d | 1e | 1f | 1g |
| 3f | 3g | 2a | 2b | 2c | 2d | 2e | 2f |
| 0  | 0  | 0  | 3a | 3b | 3c | 3d | 3e |

- six characters in six octets:

- bits number:

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
| 2g | 1a | 1b | 1c | 1d | 1e | 1f | 1g |
| 3f | 3g | 2a | 2b | 2c | 2d | 2e | 2f |
| 4e | 4f | 4g | 3a | 3b | 3c | 3d | 3e |
| 5d | 5e | 5f | 5g | 4a | 4b | 4c | 4d |
| 6c | 6d | 6e | 6f | 6g | 5a | 5b | 5c |
| 0  | 0  | 0  | 0  | 0  | 0  | 6a | 6b |

- seven characters in seven octets:

- bits number:

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
| 2g | 1a | 1b | 1c | 1d | 1e | 1f | 1g |
| 3f | 3g | 2a | 2b | 2c | 2d | 2e | 2f |
| 4e | 4f | 4g | 3a | 3b | 3c | 3d | 3e |
| 5d | 5e | 5f | 5g | 4a | 4b | 4c | 4d |
| 6c | 6d | 6e | 6f | 6g | 5a | 5b | 5c |
| 7b | 7c | 7d | 7e | 7f | 7g | 6a | 6b |
| 0  | 0  | 0  | 1  | 1  | 0  | 1  | 7a |

The bit number zero is always transmitted first.

- eight characters in seven octets:

- bits number:

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
| 2g | 1a | 1b | 1c | 1d | 1e | 1f | 1g |
| 3f | 3g | 2a | 2b | 2c | 2d | 2e | 2f |
| 4e | 4f | 4g | 3a | 3b | 3c | 3d | 3e |
| 5d | 5e | 5f | 5g | 4a | 4b | 4c | 4d |
| 6c | 6d | 6e | 6f | 6g | 5a | 5b | 5c |
| 7b | 7c | 7d | 7e | 7f | 7g | 6a | 6b |
| 8a | 8b | 8c | 8d | 8e | 8f | 8g | 7a |

- nine characters in eight octets:

- bits number:

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
| 2g | 1a | 1b | 1c | 1d | 1e | 1f | 1g |
| 3f | 3g | 2a | 2b | 2c | 2d | 2e | 2f |
| 4e | 4f | 4g | 3a | 3b | 3c | 3d | 3e |
| 5d | 5e | 5f | 5g | 4a | 4b | 4c | 4d |
| 6c | 6d | 6e | 6f | 6g | 5a | 5b | 5c |
| 7b | 7c | 7d | 7e | 7f | 7g | 6a | 6b |
| 8a | 8b | 8c | 8d | 8e | 8f | 8g | 7a |
| 0  | 9a | 9b | 9c | 9d | 9e | 9f | 9g |



- fifteen characters in fourteen octets:
- bits number:

|     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 7   | 6   | 5   | 4   | 3   | 2   | 1   | 0   |
| 2g  | 1a  | 1b  | 1c  | 1d  | 1e  | 1f  | 1g  |
| 3f  | 3g  | 2a  | 2b  | 2c  | 2d  | 2e  | 2f  |
| 4e  | 4f  | 4g  | 3a  | 3b  | 3c  | 3d  | 3e  |
| 5d  | 5e  | 5f  | 5g  | 4a  | 4b  | 4c  | 4d  |
| 6c  | 6d  | 6e  | 6f  | 6g  | 5a  | 5b  | 5c  |
| 7b  | 7c  | 7d  | 7e  | 7f  | 7g  | 6a  | 6b  |
| 8a  | 8b  | 8c  | 8d  | 8e  | 8f  | 8g  | 7a  |
| 10g | 9a  | 9b  | 9c  | 9d  | 9e  | 9f  | 9g  |
| 11f | 11g | 10a | 10b | 10c | 10d | 10e | 10f |
| 12e | 12f | 12g | 11a | 11b | 11c | 11d | 11e |
| 13d | 13e | 13f | 13g | 12a | 12b | 12c | 12d |
| 14c | 14d | 14e | 14f | 14g | 13a | 13b | 13c |
| 15b | 15c | 15d | 15e | 15f | 15g | 14a | 14b |
| 0   | 0   | 0   | 1   | 1   | 0   | 1   | 15a |

- sixteen characters in fourteen octets:
- bits number:

|     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 7   | 6   | 5   | 4   | 3   | 2   | 1   | 0   |
| 2g  | 1a  | 1b  | 1c  | 1d  | 1e  | 1f  | 1g  |
| 3f  | 3g  | 2a  | 2b  | 2c  | 2d  | 2e  | 2f  |
| 4e  | 4f  | 4g  | 3a  | 3b  | 3c  | 3d  | 3e  |
| 5d  | 5e  | 5f  | 5g  | 4a  | 4b  | 4c  | 4d  |
| 6c  | 6d  | 6e  | 6f  | 6g  | 5a  | 5b  | 5c  |
| 7b  | 7c  | 7d  | 7e  | 7f  | 7g  | 6a  | 6b  |
| 8a  | 8b  | 8c  | 8d  | 8e  | 8f  | 8g  | 7a  |
| 10g | 9a  | 9b  | 9c  | 9d  | 9e  | 9f  | 9g  |
| 11f | 11g | 10a | 10b | 10c | 10d | 10e | 10f |
| 12e | 12f | 12g | 11a | 11b | 11c | 11d | 11e |
| 13d | 13e | 13f | 13g | 12a | 12b | 12c | 12d |
| 14c | 14d | 14e | 14f | 14g | 13a | 13b | 13c |
| 15b | 15c | 15d | 15e | 15f | 15g | 14a | 14b |
| 16a | 16b | 16c | 16d | 16e | 16f | 16g | 15a |

The bit number zero is always transmitted first.

Therefore, in 160 octets, is it possible to pack  $(160 \cdot 8) / 7 = 182.8$ , that is 182 characters. The remaining 6 bits are set to zero as stated above.

Packing of 7 bit characters in USSD strings is done in the same way as for SMS (subclause 7.1.2.1). The character stream is bit padded to octet boundary with binary zeroes as shown above.

If the total number of characters to be sent equals  $(8n-1)$  where  $n = 1, 2, 3$  etc. then there are 7 spare bits at the end of the message. To avoid the situation where the receiving entity confuses 7 binary zero pad bits as the @ character, the carriage return or <CR> character (defined in subclause 7.1.1) shall be used for padding in this situation, just as for Cell Broadcast.

The sending entity may optionally use the <CR> character as an 'end of transmission' character by adding <CR> after the last USSD character followed by bit padding to octet boundary irrespective of the number of characters in the message.

If <CR> is intended to be the last character and the message (including the wanted <CR>) ends on an octet boundary, then another <CR> must be added together with a padding bit 0. The receiving entity will perform the carriage return function twice, but this will not result in misoperation as the definition of <CR> in subclause 7.1.1 is identical to the definition of <CR><CR>.

The receiving entity shall remove the final <CR> character where the message ends on an octet boundary with <CR> as the last character.

Under certain circumstances, old mobiles will perform the carriage return function after displaying the last USSD character received.

## 6.2 Alphabet tables

This clause provides tables for all the alphabets to be supported by SMS. The default alphabet is mandatory. Additional alphabets are optional. Irrespective of support of an individual alphabet, an MS shall have the ability to store a short message coded in any alphabet on the SIM.

### 6.2.1 Default alphabet

Bits per character: 7

SMS User Data Length meaning: Number of characters

CBS/USSD pad character: CR

Character table:

|    |    |    |    | b7 | 0  | 0  | 0  | 0 | 1 | 1 | 1 | 1 |
|----|----|----|----|----|----|----|----|---|---|---|---|---|
|    |    |    |    | b6 | 0  | 0  | 1  | 1 | 0 | 0 | 1 | 1 |
|    |    |    |    | b5 | 0  | 1  | 0  | 1 | 0 | 1 | 0 | 1 |
| b4 | b3 | b2 | b1 |    | 0  | 1  | 2  | 3 | 4 | 5 | 6 | 7 |
| 0  | 0  | 0  | 0  | 0  | @  | Δ  | SP | 0 | i | P | ı | p |
| 0  | 0  | 0  | 1  | 1  | £  | _  | !  | 1 | A | Q | a | q |
| 0  | 0  | 1  | 0  | 2  | \$ | Φ  | "  | 2 | B | R | b | r |
| 0  | 0  | 1  | 1  | 3  | ¥  | Γ  | #  | 3 | C | S | c | s |
| 0  | 1  | 0  | 0  | 4  | è  | Λ  | α  | 4 | D | T | d | t |
| 0  | 1  | 0  | 1  | 5  | é  | Ω  | %  | 5 | E | U | e | u |
| 0  | 1  | 1  | 0  | 6  | ù  | Π  | &  | 6 | F | V | f | v |
| 0  | 1  | 1  | 1  | 7  | î  | Ψ  | '  | 7 | G | W | g | w |
| 1  | 0  | 0  | 0  | 8  | ò  | Σ  | (  | 8 | H | X | h | x |
| 1  | 0  | 0  | 1  | 9  | ç  | Θ  | )  | 9 | I | Y | i | y |
| 1  | 0  | 1  | 0  | 10 | LF | Ξ  | *  | : | J | Z | j | z |
| 1  | 0  | 1  | 1  | 11 | Ø  | 1) | +  | ; | K | Ä | k | ä |
| 1  | 1  | 0  | 0  | 12 | ø  | Æ  | ,  | < | L | Ö | l | ö |
| 1  | 1  | 0  | 1  | 13 | CR | æ  | -  | = | M | Ñ | m | ñ |
| 1  | 1  | 1  | 0  | 14 | Å  | ß  | .  | > | N | Ü | n | ü |
| 1  | 1  | 1  | 1  | 15 | å  | É  | /  | ? | O | § | o | à |

## History

| Document history |   |
|------------------|---|
| December 1995    | Publication of GTS 03.38 version 5.0.0                      |
| March 1996       | Publication of GTS 03.38 version 5.1.0                      |
| May 1996         | Publication of GTS 03.38 version 5.2.0                      |
| July 1996        | Publication of GTS 03.38 version 5.3.0                      |
| November 1996    | Unified Approval Procedure UAP 58: 1996-11-18 to 1997-03-14 |
| April 1997       | First Edition   |