INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

T.85

Corrigendum 1
(02/97)

SERIES T: TERMINALS FOR TELEMATIC SERVICES

Application profile for Recommendation T.82 – Progressive bi-level image compression (JBIG coding scheme) for facsimile apparatus

Corrigendum 1

ITU-T Recommendation T.85 – Corrigendum 1

(Previously CCITT Recommendation)

ITU-T T-SERIES RECOMMENDATIONS

TERMINALS FOR TELEMATIC SERVICES

For further details, please refer to ITU-T List of Recommendations.

FOREWORD

The ITU-T (Telecommunication Standardization Sector) is a permanent organ of the International Telecommunication Union (ITU). The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, 1993, revised Geneva, 1996).

Corrigendum 1 to ITU-T Recommendation T.85 was prepared by ITU-T Study Group 8 (1993-1996) and was approved on the 13th of February 1997.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

CONTENTS

Corrigendum 1 to Recommendation T.85

APPLICATION PROFILE FOR RECOMMENDATION T.82 – PROGRESSIVE BI-LEVEL IMAGE COMPRESSION (JBIG CODING SCHEME) FOR FACSIMILE APPARATUS

CORRIGENDUM 1

(Geneva, 1997)

1) Modify 4.4 and add a new subclause 4.5 as follows:

4.4 Usage of the marker segment

COMMENT – The interpretation of the information transmitted by COMMENT marker segment depends on the implementation of both ends.

NEWLEN – To enable a page length to be changed using a T.82 NEWLEN marker in front of a "null" stripe (e.g. in BASIC mode-2 or OPTION mode):

1) The NEWLEN marker in front of a "null" stripe shall be the first marker in front of any such "null" stripe.

2) The T.82 header bit VLENGTH must be set to "1" and the decoder must look beyond the SDRST/SDNORM at the end of all stripes, in order to find any immediately following NEWLEN marker that is in front of a "null" stripe. If such a NEWLEN marker is found, all previous values of L0 and Y_D are to be cleared and Y_D set to equal NEWLEN.

4.5 Initialization of the arithmetic coder/decoder and adaptive template's status

At the top of each page of facsimile transmission, the state of arithmetic coder/decoder and adaptive template shall be initialized.

2) Modify Appendix I as follows:

Appendix I

Examples of the usage of NEWLEN marker segment

This Appendix describes examples of the usage of NEWLEN marker segment for the case when the facsimile terminal cannot identify the vertical size Y_D of the page to be transmitted when it starts coding. This Appendix applies to the single-progression sequential coding described in clause 4.

The examples shown in this Appendix illustrate the application of 6.2.6.2/T.82.

I.1 BASIC mode

The first two examples show the case when one page of 500 lines is transmitted in BASIC mode with L0 = 128. One page is coded into multiple stripes with the condition that the vertical length of the page is unknown when the transmitting facsimile terminal starts coding. For the first example, Y_D is set to 0xffffffff. Then, at the line of 500, the image data exhausts.

Data stream for BASIC mode-1

BIH (Y_D = 0xffffffff, L0 = 128, VLENGTH = 1 – Other parameters shall be set appropriately).

Encoded image data of the first stripe (line 1-128).

ESC, SDNORM.

Encoded image data of the second stripe (line 129-256).

ESC, SDNORM.

Encoded image data of the third stripe (line 257-384).

ESC, SDNORM.

ESC, NEWLEN, New Y_D (= 500).

Encoded image data of the fourth stripe (line 385-500).

ESC, SDNORM or SDRST.

The following data stream is an example, where the length of the page is not known before coding the last stripe. Note that this case requires the use of a "null" stripe. This example will also illustrate an estimate of page length that is not the maximum possible (e.g. Y_D = 1024 = 0x00000400).

Data stream for BASIC mode-2

BIH (Y_D = 0x00000400, L0 = 128, VLENGTH =1 – Other parameters shall be set appropriately).

Encoded image data of the first stripe (line 1-128).

ESC, SDNORM.

Encoded image data of the second stripe (line 129-256).

ESC, SDNORM.

Encoded image data of the third stripe (line 257-384).

ESC, SDNORM.

Encoded image data of the fourth stripe (line 385-500).

ESC, SDNORM.

ESC, NEWLEN (note that this is first marker), New Y_D (= 500).

Encoded data of the fifth stripe (note that "null" means there is no data).

ESC, SDNORM or SDRST.

I.2 OPTION mode

This example shows the case of "one stripe per page" (also requires an "added" null stripe) transmission in OPTION mode. This optional mode can only be used following a successful negotiation. Note that a coder or decoder that cannot support the actual stripe size may have to terminate the call. At the beginning of encoding, as the vertical length of the page is undetermined, Y_D is set to the maximum value, Y_D = 0xffffffff (one possible choice). As the stripe size is also undetermined, L0 is set to the same value as Y_D, L0 = 0xffffffff. The actual number of the vertical lines is 500.

Data stream for OPTION mode

BIH (Y_D = 0xffffffff, L0 = 0xffffffff, VLENGTH =1 – Other parameters shall be set appropriately).

Encoded image data of the first stripe (line 1-500).

ESC, SDNORM.

ESC, NEWLEN (note that this is first marker), New Y_D (= 500).

Encoded data of the second stripe (note that "null" means there is no data).

ESC, SDNORM or SDRST.

3) Add Appendix II as follows:

Appendix II

The bit order of Coded Data Transmission on communication line conforming to the description of clause 3 “Principle” of this Recommendation

Here, the coded data stream example described in 7.1/T.82 and Table 26/T.82 is used as an example.

PSCD: 6989 995c 32ea faa0 -------

The bit order of the coded data at the output of encoder is described as

MSB LSB

01101001	10001001	10011001	01011100	00110010	11101010
69	89	99	5c	32	ea

These coded data are transmitted LSB first for each byte, then

10010110	1001001	10011001	00111010	01001100	01010111

is the bit order on communication line.

The receiving FAX will reorder the received bits and treat the coded data as

MSB LSB

01101001	10001001	10011001	01011100	00110010	11101010
69	89	99	5c	32	ea

Then the data is to be decoded according to the algorithm described in Recommendation T.82.

ITU-T RECOMMENDATIONS SERIES

Series A	Organization of the work of the ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	Maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	**Terminals for telematic services**
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks and open system communication
Series Z	Programming languages