TYPE: Coding system different from that of ISO 2022 with Standard return	REGISTRATION NUMBER: 178 DATE OF REGISTRATION: 93-01-21 amended 95-06-08

ESCAPE SEQUENCE: ESC 02/05 04/02

NAME UCS Transformation Format One (UTF-1)

DESCRIPTION

This format transforms the coded representation of graphic characters in ISO/IEC 10646 into a form that does not use octet values specified in ISO/IEC 2022 as coded representations of C0, SPACE, DEL, or C1 characters

A detailed specification of this format is attached here.

SPONSOR ISO/IEC JTC 1/SC2/WG2

ORIGIN ISO/IEC 10646, First edition 1993.

FIELD OF UTILISATION

This format permits text data that conforms to ISO/IEC 10646 to be transmitted through communication systems which are sensitive to octet values for control characters coded according to the structure of ISO/IEC 2022.

UCS Transformation Format One (UTF-1)

The following method transforms the coded representation of graphic characters in the coded character set of ISO/IEC 10646 into a form that does not use octet values specified in ISO 2022 as coded representations of C0, SPACE, DEL, or C1 characters, and can thus be used for transmitting text data through communication systems that are sensitive to these octet values.

This specification is taken from Annex G of the First edition (1993) of ISO/IEC 10646-1. Definitions of the terms used here will be found in ISO/IEC 10646.

1 Outline of the algorithm

The algorithm can be summarized as follows:

1. A UCS character from 0000 0000 to 0000 009F is mapped to the corresponding octet from 00 to 9F.

2. A UCS character from 0000 00A0 to 0000 00FF is mapped to a sequence of two octets, with the first octet being A0, and the second octet in the range A0 to FF.

3. A UCS character from 0000 0100 to 0000 4015 is mapped to a sequence of two octets, with the first octet in the range from A1 to F5, and the second octet having 190 values in the range 21 to 7E or the range A0 to FF.

4. A UCS character from 0000 4016 to 0003 8E2D is mapped to a sequence of three octets, with the first octet in the range from F6 to FB, and the other octets in the range 21 to 7E or the range A0 to FF.

5. A UCS character at 0003 8E2E or larger is mapped to a sequence of five octets, with the first octet in the range from FC to FF, and the other octets in the range 21 to 7E or the range A0 to FF.

Notice that four-octet sequences are not used, since this maximizes the number of characters that can use the three-octet form.

2 Notation

1. All numbers are in hexadecimal notation.

2. Octet boundaries in the transformed text are indicated with semicolons.

3. The symbol "%" indicates the modulo operation, e.g.:

   x % y = x modulo y

   The symbol "/" indicates the integer division operation, e.g.:

   7 / 3 = 2

   Superscripting indicates the power-of operation, e.g.:

   $2^3 = 8$

   Precedence is "^" > "/" > "%", e.g.:

   $x / y^2 \% w = ((x / (y^2)) \% w)$

4. T(z) is defined for z = 00..FF such that

z = 00 .. 5D:	T(z) = z + 21
z = 5E .. BD:	T(z) = z + 42
z = BE .. DE:	T(z) = z - BE
z = DF .. FF:	T(z) = z - 60

   e.g. T(00) = 21, T(5D) = 7E,
   T(5E) = A0, T(BD) = FF,
   T(BE) = 00, T(DE) = 20,
   T(DF) = 7F, T(FF) = 9F

5. U(z) is the inverse of T(z): that is, U(T(z)) = z, and T(U(z)) = z:

z = 00 .. 20:	U(z) = z + BE
z = 21 .. 7E:	U(z) = z - 21
z = 7F .. 9F:	U(z) = z + 60
z = A0 .. FF:	U(z) = z - 42

   e.g. U(00) = BE, U(20) = DE,
   U(21) = 00, U(7E) = 5D,
   U(7F) = DF, U(9F) = FF,
   U(A0) = 5E, U(FF) = BD

6. The algorithm in this annex has been presented in a descriptive format. The implementation may differ for efficiency. For example, the T and U functions can be implemented with a small table lookup.

3 From UCS to UTF-1 format

Condition/UCS	UTF-1 octets
x = 0000 0000 .. 0000 009F:	x;
x = 0000 00A0 .. 0000 00FF:	A0; x;
x = 0000 0100 .. 0000 4015: (y = x - 0000 0100)	A1 + y / BE; T(y % BE);
x = 0000 4016 .. 0003 8E2D: (y = x - 0000 4016)	F6 + y / BE^2; T(y / BE % BE); T(y % BE);
x = 0003 8E2E .. 7FFF FFFF: (y = x - 0003 8E2E)	FC + y / BE^4; T(y / BE^3 % BE); T(y / BE^2 % BE); T(y / BE % BE); T(y % BE);

4 From UTF-1 to UCS format

Condition/UTF-1	UCS
x = 00 .. 9F;	x
x = A0; y;	y
x = A1 .. F5; y;	(x - A1) × BE + U(y) +0000 0100
x = F6 .. FB; y; z;	(x - F6) × BE2 + U(y) × BE + U(z) + 0000 4016
x = FC .. FF; y; z; v; w;	(x - FC) × BE^4 + U(y) × BE^3 + U(z) × BE^2 + U(v) × BE + U(w) + 0003 8E2E

5 Identification of UTF-1

When the escape sequences from ISO/IEC 2022 are used, the identification of the UTF-1 can be given by a designation sequence:

ESC 02/05 04/02

If such an escape sequence appears within a CC-data-element conforming to ISO/IEC 2022, it shall consist only of the sequences of bit combinations as shown above.

If such an escape sequence appears within a CC-data-element conforming to ISO/IEC 10646, it shall be padded in accordance with clause 16 of ISO/IEC 10646.

When the escape sequences from ISO/IEC 2022 are used, the identification of the return from UTF-1 to the coding system of ISO 2022 shall be by the escape sequence ESC 02/05 04/00.