

ISO/IEC JTC 1/SC 2
Coded Character Sets
Secretariat: [Japan \(JISC\)](#)

DOC. TYPE	Working Draft Text (for information or comment)	
TITLE	Working Draft for Merged ISO/IEC 10646, Information technology -- Universal Multiple-Octet Coded Character Set (UCS) -- Architecture and Basic Multilingual Plane, Supplementary Planes [WG 2 N 2499R]	
SOURCE	Project editor	
PROJECT	JTC 1.02.18	
STATUS	In anticipation of SC 2 and JTC 1 approval of SC 2 N 3638, this working draft is circulated to the SC 2 members for comment by 2002-11-22. Experts from SC 2 NBs and liaison organizations are requested to provide input and feedback on this new version of the working draft of the merged document instead of the original WG 2 N 2499 as many typos were corrected and other editorial content was refined by the project editor.	
ACTION ID	COM	
DUE DATE	2002-11-22	
DISTRIBUTION	P, O and L Members of ISO/IEC JTC 1/SC 2 ; ISO/IEC JTC 1 Secretariat; ISO/IEC ITTF	
ACCESS LEVEL	Open	
ISSUE NO.	150	
FILE	NAME	02n3639.pdf
	SIZE (KB)	96
	PAGES	

Secretariat ISO/IEC JTC 1/SC 2 - IPSJ/ITSCJ *(Information Processing Society of Japan/Information Technology Standards Commission of Japan) Room 308-3, Kikai-Shinko-Kaikan Bldg., 3-5-8, Shiba-Koen, Minato-ku, Tokyo 105-0011 Japan *Standard Organization Accredited by JISC
 Telephone: +81-3-3431-2808; Facsimile: +81-3-3431-6493; E-mail: kimura@itscj.ipsj.or.jp

SC2/WG2 N2499R

ISO/IEC International Standard

Working Draft International Standard 10646 3rd Edition

ISO/IEC WD 10646 3rd Edition

2002-09-20

**Information technology — Universal Multiple-Octet
Coded Character Set (UCS) —**

Architecture and Basic Multilingual Plane

Supplementary Planes

Reserved for final ISO Copyright statement

Contents		Page
1	Scope	1
2	Conformance.....	1
3	Normative references	2
4	Terms and definitions	2
5	General structure of the UCS.....	4
6	Basic structure and nomenclature.....	5
7	General requirements for the UCS.....	9
8	The Basic Multilingual Plane	9
9	Supplementary planes	10
10	Private use groups, planes, and zones	10
11	Revision and updating of the UCS	10
12	Subsets	10
13	Coded representation forms of the UCS	11
14	Implementation levels	11
15	Use of control functions with the UCS.....	11
16	Declaration of identification of features	12
17	Structure of the code tables and lists	13
18	Block names.....	13
19	Characters in bi-directional context.....	14
20	Special characters.....	14
21	Presentation forms of characters	17
22	Compatibility characters.....	18
23	Order of characters	18
24	Normalization forms	18
25	Combining characters	18
26	Special features of individual scripts	20
27	Source references for CJK Ideographs.....	20
28	Character names and annotations	22
29	Structure of the Basic Multilingual Plane.....	25
30	Structure of the Supplementary Multilingual Plane for Scripts and symbols....	27
31	Structure of the Supplementary Ideographic Plane	28
32	Supplementary Special-purpose Plane.....	28
33	Code tables and lists of character names	28

Annexes

A	Collections of graphic characters for subsets	1001
B	List of combining characters	1011
C	Transformation format for 16 planes of Group 00 (UTF-16)	1017

D	UCS Transformation Format 8 (UTF-8)	1020
E	Mirrored characters in Arabic bi-directional context	1024
F	Alternate format characters	1027
G	Alphabetically sorted list of character names	1032
H	The use of “signatures” to identify UCS	1033
J	Recommendation for combined receiving/originating devices with internal storage	1034
K	Notations of octet value representations	1035
L	Character naming guidelines	1036
M	Sources of characters	1038
N	External references to character repertoires	1042
P	Additional information on characters	1044
Q	Code mapping table for Hangul syllables	1047
R	Names of Hangul syllables	1048
S	Procedure for the unification and arrangement of CJK Ideographs	1049
T	Language tagging using Tag Characters	1057
U	Usage of musical symbols	1059

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields or technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC1. Draft international Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75% of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of ISO/IEC 10646 may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 10646 was prepared by Joint Technical Committee ISO/IEC JTC1, *Information technology*, Subcommittee SC 2, *Coded Character sets*.

This third edition cancels and replaces the previous editions of this International Standard which were published in two parts: Part 1 second edition (ISO/IEC 10646-1:2000) and Part 2 first edition (ISO/IEC 10646-2:2001). It also incorporates Amendments 1 and 2 to Part 1 and Amendment 1 to Part 2.

Annexes A to D form a normative part of ISO/IEC 10646. Annexes E to U are for information only.

The standard contains material which may only be available to users who obtain their copy in a machine readable format. That material consists of the following printable files:

- CJKUA_SR.txt
- CJKC0SR.txt
- Allnames.txt
- HanguIX.txt
- HanguISy.txt

Introduction

ISO/IEC 10646 specifies the Universal Multiple-Octet Coded Character Set (UCS). It is applicable to the representation, transmission, interchange, processing, storage, input and presentation of the written form of the languages of the world as well as additional symbols.

By defining a consistent way of encoding multilingual text it enables the exchange of data internationally. The information technology industry gains data stability, greater global interoperability and data interchange. ISO/IEC 10646 has been widely adopted in new Internet protocols and implemented in modern operating systems and computer languages. This edition covers over 95 000 characters from the world's scripts.

Information technology — Universal Multiple-Octet Coded Character Set (UCS) —

1 Scope

ISO/IEC 10646 specifies the Universal Multiple-Octet Coded Character Set (UCS). It is applicable to the representation, transmission, interchange, processing, storage, input, and presentation of the written form of the languages of the world as well as of additional symbols.

This document:

- specifies the architecture of ISO/IEC 10646,
- defines terms used in ISO/IEC 10646,
- describes the general structure of the coded character set;
- specifies the Basic Multilingual Plane (BMP) of the UCS,
- specifies supplementary planes of the UCS: the Supplementary Multilingual Plane (SMP), the Supplementary Ideographic Plane (SIP) and the Supplementary Special-purpose Plane (SSP),
- defines a set of graphic characters used in scripts and the written form of languages on a world-wide scale;
- specifies the names for the graphic characters of the BMP, SMP, SIP, SSP and their coded representations;
- specifies the four-octet (32-bit) canonical form of the UCS: UCS-4;
- specifies a two-octet (16-bit) BMP form of the UCS: UCS-2;
- specifies the coded representations for control functions;
- specifies the management of future additions to this coded character set.

The UCS is a coding system different from that specified in ISO/IEC 2022. The method to designate UCS from ISO/IEC 2022 is specified in 16.2.

Graphic characters that are already encoded in the BMP are not duplicated in the supplementary planes. In addition, any character is assigned to only one code position within the set of supplementary planes.

NOTE 1 – The Unicode Standard, Version 4.0 includes a set of characters, names, and coded representations that are identical with those in this International Standard. It additionally provides details of character properties, process-

ing algorithms, and definitions that are useful to implementers.

NOTE 2 – Previous editions of ISO/IEC 10646 were published in parts: Part 1 specified the architecture and the BMP, Part 2 specified the SMP, SIP and SSP.

2 Conformance

2.1 General

Whenever private use characters are used as specified in ISO/IEC 10646, the characters themselves shall not be covered by these conformance requirements.

2.2 Conformance of information interchange

A coded-character-data-element (CC-data-element) within coded information for interchange is in conformance with ISO/IEC 10646 if

- a) all the coded representations of graphic characters within that CC-data-element conform to clauses 6 and 7, to an identified form chosen from clause 13 or annex C or annex D, and to an identified implementation level chosen from clause 14;
- b) all the graphic characters represented within that CC-data-element are taken from those within an identified subset (clause 12);
- c) all the coded representations of control functions within that CC-data-element conform to clause 15.

A claim of conformance shall identify the adopted form, the adopted implementation level and the adopted subset by means of a list of collections and/or characters.

2.3 Conformance of devices

A device is in conformance with ISO/IEC 10646 if it conforms to the requirements of item a) below, and either or both of items b) and c).

NOTE – The term device is defined (in 4.18) as a component of information processing equipment which can transmit and/or receive coded information within CC-data-elements. A device may be a conventional input/output device, or a process such as an application program or gateway function.

A claim of conformance shall identify the document that contains the description specified in a) below, and shall identify the adopted form(s), the adopted implementation level, the adopted subset (by means of a list of collections and/or characters), and the selection of

control functions adopted in accordance with clause 15.

- a) **Device description:** A device that conforms to ISO/IEC 10646 shall be the subject of a description that identifies the means by which the user may supply characters to the device and/or may recognize them when they are made available to the user, as specified respectively, in sub-clauses b), and c) below.
- b) **Originating device:** An originating device shall allow its user to supply any characters from an adopted subset, and be capable of transmitting their coded representations within a CC-data-element in accordance with the adopted form and implementation level.
- c) **Receiving device:** A receiving device shall be capable of receiving and interpreting any coded representation of characters that are within a CC-data-element in accordance with the adopted form and implementation level, and shall make any corresponding characters from the adopted subset available to the user in such a way that the user can identify them.

Any corresponding characters that are not within the adopted subset shall be indicated to the user. The way used for indicating them need not distinguish them from each other.

NOTE 1 – An indication to the user may consist of making available the same character to represent all characters not in the adopted subset, or providing a distinctive audible or visible signal when appropriate to the type of user.

NOTE 2 – See also annex J for receiving devices with re-transmission capability.

3 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of ISO/IEC 10646. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on ISO/IEC 10646 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO/IEC 2022:1994 *Information technology — Character code structure and extension techniques.*

ISO/IEC 6429:1992 *Information technology — Control functions for coded character sets.*

Unicode Standard Annex, UAX#9, The Unicode Bidirectional Algorithm, Version 3.2.0, 2002-03-27.

Unicode Standard Annex, UAX#15, Unicode Normalization Forms, Version 3.2.0, 2002-03-27.

4 Terms and definitions

For the purposes of ISO/IEC 10646, the following terms and definitions apply:

4.1 Basic Multilingual Plane (BMP):

Plane 00 of Group 00.

4.2 Block:

A contiguous range of code positions to which a set of characters that share common characteristics, such as a script, are allocated. A block does not overlap another block. One or more of the code positions within a block may have no character allocated to them.

4.3 Canonical form:

The form with which characters of this coded character set are specified using four octets to represent each character.

4.4 CC-data-element (coded-character-data-element):

An element of interchanged information that is specified to consist of a sequence of coded representations of characters, in accordance with one or more identified standards for coded character sets.

4.5 Cell:

The place within a row at which an individual character may be allocated.

4.6 Character:

A member of a set of elements used for the organization, control, or representation of data.

4.7 Character boundary:

Within a stream of octets the demarcation between the last octet of the coded representation of a character and the first octet of that of the next coded character.

4.8 Coded character:

A character together with its coded representation.

4.9 Coded character set:

A set of unambiguous rules that establishes a character set and the relationship between the characters of the set and their coded representation.

4.10 Code table:

A table showing the characters allocated to the octets in a code.

4.11 Collection:

A set of coded characters which is numbered and named and which consists of those coded characters whose code positions lie within one or more identified ranges.

NOTE – If any of the identified ranges include code positions to which no character is allocated, the repertoire of the collection will change if an additional character is assigned to any of those positions at a future amendment of this International Standard. However it is intended that the collection number and name will remain unchanged in future editions of this International Standard.

4.12 Combining character:

A member of an identified subset of the coded character set of ISO/IEC 10646 intended for combination with the preceding non-combining graphic character, or with a sequence of combining characters preceded by a non-combining character (see also 4.14).

NOTE – ISO/IEC 10646 specifies several subset collections which include combining characters.

4.13 Compatibility character:

A graphic character included as a coded character of ISO/IEC 10646 primarily for compatibility with existing coded character sets.

4.14 Composite sequence:

A sequence of graphic characters consisting of a non-combining character followed by one or more combining characters (see also 4.12).

NOTE 1 – A graphic symbol for a composite sequence generally consists of the combination of the graphic symbols of each character in the sequence.

NOTE 2 – A composite sequence is not a character and therefore is not a member of the repertoire of ISO/IEC 10646.

4.15 Control function:

An action that affects the recording, processing, transmission, or interpretation of data, and that has a coded representation consisting of one or more octets.

4.16 Default state:

The state that is assumed when no state has been explicitly specified.

4.17 Detailed code table:

A code table showing the individual characters, and normally showing a partial row.

4.18 Device:

A component of information processing equipment which can transmit and/or receive coded information within CC-data-elements. (It may be an input/output device in the conventional sense, or a process such as an application program or gateway function.)

4.19 Fixed collection:

A collection in which every code position within the identified range(s) has a character allocated to it, and which is intended to remain unchanged in future editions of this International Standard.

4.20 Graphic character:

A character, other than a control function, that has a visual representation normally handwritten, printed, or displayed.

4.21 Graphic symbol:

The visual representation of a graphic character or of a composite sequence.

4.22 Group:

A subdivision of the coding space of this coded character set; of 256 x 256 x 256 cells.

4.23 High-half zone:

a set of cells reserved for use in UTF-16 (see annex C); an RC-element corresponding to any of these cells may be used in UTF-16 as the first of a pair of RC-elements which represents a character from a plane other than the BMP.

4.24 Interchange:

The transfer of character coded data from one user to another, using telecommunication means or interchangeable media.

4.25 Interworking:

The process of permitting two or more systems, each employing different coded character sets, meaningfully to interchange character coded data; conversion between the two codes may be involved.

4.26 ISO/IEC 10646-1

A former subdivision of the standard. It is also referred to as Part 1 of ISO/IEC 10646 and contained the specification of the overall architecture and the Basic Multilingual Plane (BMP). There are a First and a Second Edition of ISO/IEC 10646-1.

4.27 ISO/IEC 10646-2

A former subdivision of the standard. It is also referred to as Part 2 of ISO/IEC 10646 and contained the specification of the Supplementary Multilingual Plane (SMP), the Supplementary Ideographic Plane (SIP) and the Supplementary Special-purpose Plane (SSP). There is only a First Edition of ISO/IEC 10646-2.

4.28 Low-half zone:

A set of cells reserved for use in UTF-16 (see annex C); an RC-element corresponding to any of these cells may be used in UTF-16 as the second of a pair of RC-elements which represents a character from a plane other than the BMP.

4.29 Octet:

An ordered sequence of eight bits considered as a unit.

4.30 Plane:

A subdivision of a group; of 256 x 256 cells.

4.31 Presentation; to present:

The process of writing, printing, or displaying a graphic symbol.

4.32 Presentation form:

In the presentation of some scripts, a form of a graphic symbol representing a character that depends on the position of the character relative to other characters.

4.33 Private use plane:

A plane within this coded character set the contents of which is not specified in ISO/IEC 10646 (see clause 10)

4.34 RC-element:

A two-octet sequence comprising the R-octet and the C-octet (see 6.2) from the four octet sequence (in the canonical form) that corresponds to a cell in the coding space of this coded character set.

4.35 repertoire:

A specified set of characters that are represented in a coded character set.

4.36 row:

A subdivision of a plane; of 256 cells.

4.37 script:

A set of graphic characters used for the written form of one or more languages.

4.38 Supplementary plane:

A plane that accommodates characters which have not been allocated to the Basic Multilingual Plane.

4.39 Supplementary Multilingual Plane for scripts and symbols (SMP)

Plane 01 of Group 00.

4.40 Supplementary Ideographic Plane (SIP)

Plane 02 of Group 00.

4.41 Supplementary Special-purpose Plane (SSP)

Plane 0E of Group 00.

4.42 Unpaired RC-element:

An RC-element in a CC-data element that is either:

- an RC-element from the high-half zone that is not immediately followed by an RC-element from the low-half zone, or
- an RC-element from the low-half zone that is not immediately preceded by a high-half RC-element from the high-half zone.

4.43 User:

A person or other entity that invokes the service provided by a device. (This entity may be a process such as an application program if the “device” is a code converter or a gateway function, for example.)

4.44 Zone:

A sequence of cells of a code table, comprising one or more rows, either in whole or in part, containing characters of a particular class (for example see clause 8).

5 General structure of the UCS

The general structure of the Universal Multiple-Octet Coded Character Set (referred to hereafter as “this coded character set”) is described in this explanatory

clause, and is illustrated in figures 1 and 2. The normative specification of the structure is given in the following clauses.

The value of any octet is expressed in hexadecimal notation from 00 to FF in ISO/IEC 10646 (see annex K).

The canonical form of this coded character set – the way in which it is to be conceived – uses a four-dimensional coding space, regarded as a single entity, consisting of 128 three-dimensional groups.

NOTE 1 – Thus, bit 8 of the most significant octet in the canonical form of a coded character can be used for internal processing purposes within a device as long as it is set to zero within a conforming CC-data-element.

Each group consists of 256 two-dimensional planes. Each plane consists of 256 one-dimensional rows, each row containing 256 cells. A character is located and coded at a cell within this coding space or the cell is declared unused.

In the canonical form, four octets are used to represent each character, and they specify the group, plane, row and cell, respectively. The canonical form consists of four octets since two octets are not sufficient to cover all the characters in the world, and a 32-bit representation follows modern processor architectures.

The four-octet canonical form can be used as a four-octet coded character set, in which case it is called UCS-4.

NOTE 2 – The use of the term “canonical” for this form does not imply any restriction or preference for this form over transformation formats that a conforming implementation may choose for the representation of UCS characters.

ISO/IEC 10646 defines graphic characters and their coded representation for the following planes:

- The Basic Multilingual Plane (BMP, Plane 00 of Group 0). The Basic Multilingual Plane can be used as a two-octet coded character set identified as UCS-2.
- The Supplementary Multilingual Plane for scripts and symbols (SMP, Plane 01 of Group 00).
- The Supplementary Ideographic Plane (SIP, Plane 02 of Group 00).
- The Supplementary Special-purpose Plane (SSP, Plane 0E of Group 0).

Additional supplementary planes may be defined in the future to accommodate additional graphic characters.

The planes that are reserved for private use are specified in clause 10. The contents of the cells in private use planes and zones are not specified in ISO/IEC 10646.

Each character is located within the coded character set in terms of its Group-octet, Plane-octet, Row-octet, and Cell-octet.

Subsets of the coding space may be used in order to give a sub-repertoire of graphic characters.

A UCS Transformation Format (UTF-16) is specified in annex C which can be used to represent characters from 16 planes of group 00, additional to the BMP, in a form that is compatible with the two-octet BMP form.

Another UCS Transformation Format (UTF-8) is specified in annex D which can be used to transmit text data through communication systems which are sensitive to octet values for control characters coded according to the 8-bit structure of ISO/IEC 2022, and to ISO/IEC 4873. UTF-8 also avoids the use of octet values according to ISO/IEC 4873 which have special significance during the parsing of file-name character strings in widely-used file-handling systems.

6 Basic structure and nomenclature

6.1 Structure

The Universal Multiple-Octet Coded Character Set as specified in ISO/IEC 10646 shall be regarded as a single entity.

This entire coded character set shall be conceived of as comprising 128 groups of 256 planes. Each plane shall be regarded as containing 256 rows of characters, each row containing 256 cells. In a code table representing the contents of a plane (such as in figure 2), the horizontal axis shall represent the least significant octet, with its smaller value to the left; and the vertical axis shall represent the more significant octet, with its smaller value at the top.

Each axis of the coding space shall be coded by one octet. Within each octet the most significant bit shall be bit 8 and the least significant bit shall be bit 1. Accordingly, the weight allocated to each bit shall be

bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1
128	64	32	16	8	4	2	1

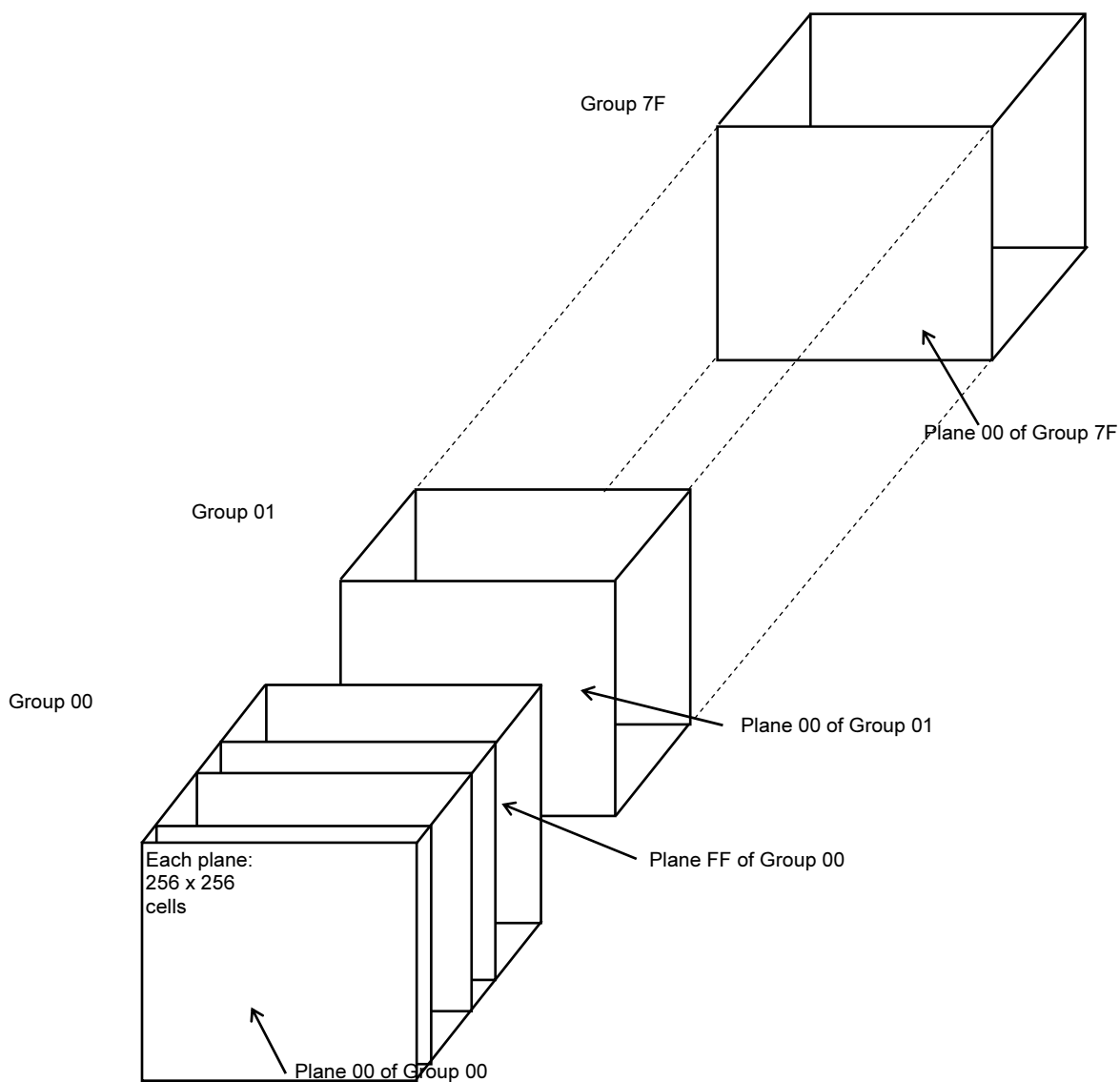
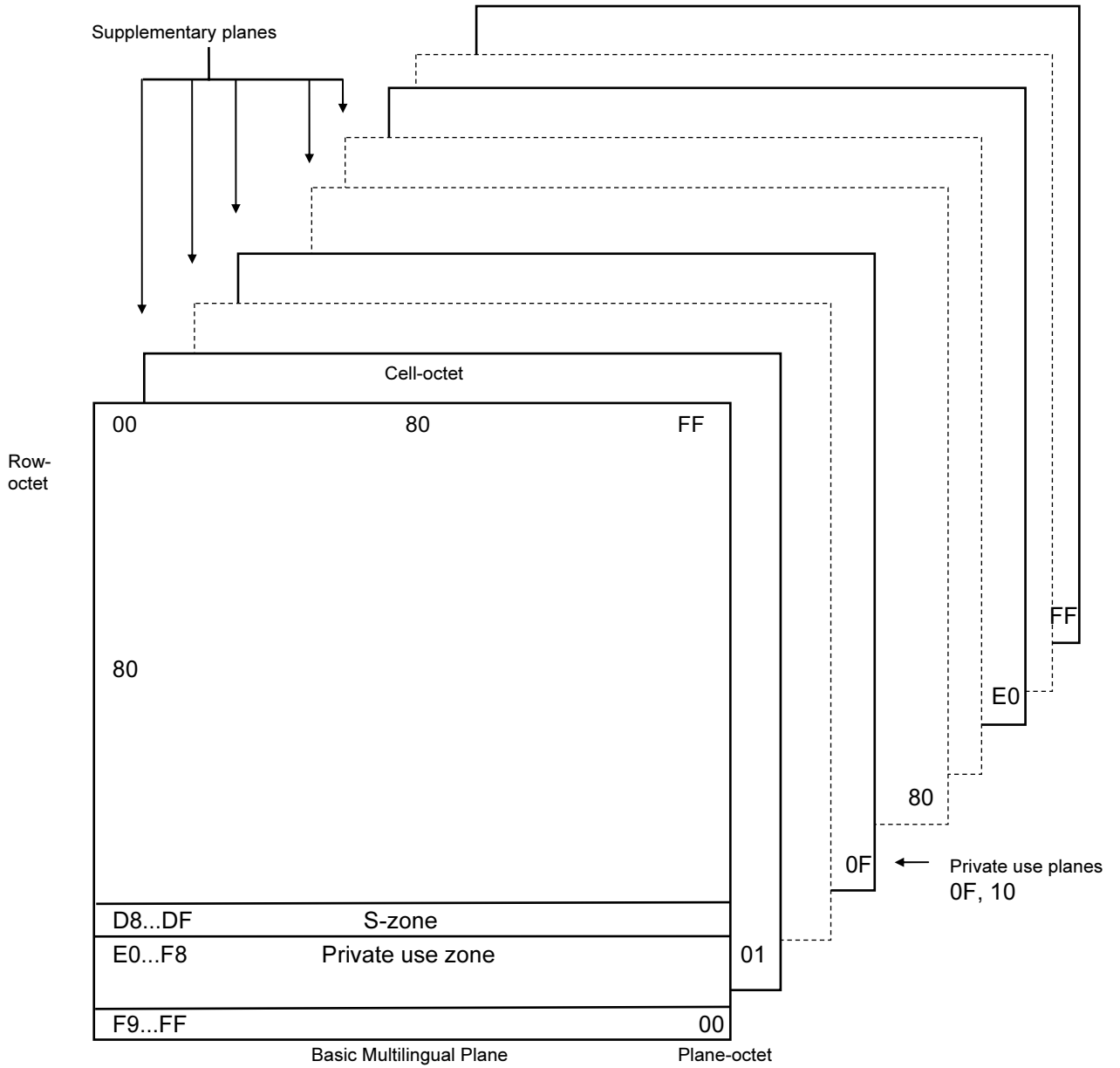


Figure 1 - Entire coding space of the Universal Multiple-Octet Coded Character Set



NOTE - Labels "S-zone" and "Private use zone" are specified in clause 8.

Figure 2 - Group 00 of the Universal Multiple-Octet Coded Character Set

6.2 Coding of characters

In the canonical form of the coded character set, each character within the entire coded character set shall be represented by a sequence of four octets. The most significant octet of this sequence shall be the group-octet. The least significant octet of this sequence shall be the cell-octet. Thus this sequence may be represented as

m.s.				l.s.
Group-octet	Plane-octet	Row-octet	Cell-octet	

where m.s. means the most significant octet, and l.s. means the least significant octet.

For brevity, the octets may be termed

m.s.				l.s.
G-octet	P-octet	R-octet	C-octet	

Where appropriate, these may be further abbreviated to G, P, R, and C.

The value of any octet shall be represented by two hexadecimal digits, for example: 31 or FE. When a single character is to be identified in terms of the values of its group, plane, row, and cell, this shall be represented such as:

0000 0030 for DIGIT ZERO

0000 0041 for LATIN CAPITAL LETTER A

When referring to characters within an identified plane, the leading four digits (for G-octet and P-octet) may be omitted. For example, within plane 00, 0030 may be used to refer to DIGIT ZERO.

When referring to characters within planes 00 to 0F, the leading three digits may be omitted. For example, the five-digit value 11100 corresponds to the canonical form 0001 1100 and the corresponding coded character is part of plane 01.

6.3 Octet order

The sequence of the octets that represent a character, and the most significant and least significant ends of it, shall be maintained as shown above. When serialized as octets, a more significant octet shall precede less significant octets. When not serialized as octets, the order of octets may be specified by agreement between sender and recipient (see 16.1 and annex H).

6.4 Naming of characters

ISO/IEC 10646 assigns a unique name to each character. The name of a character either:

- a. denotes the customary meaning of the character, or

- b. describes the shape of the corresponding graphic symbol, or
- c. follows the rule given in clause 27 for Chinese /Japanese/Korean (CJK) unified ideographs.

Guidelines to be used for constructing the names of characters in cases a. and b. are given in annex L.

6.5 Short identifiers for code positions (UIDs)

ISO/IEC 10646 defines short identifiers for each code position, including code positions that are reserved. A short identifier for any code position is distinct from a short identifier for any other code position. If a character is allocated at a code position, a short identifier for that code position can be used to refer to the character allocated at that code position.

NOTE 1 – For instance, U+DC00 identifies a code position that is permanently reserved for UTF-16, and U+FFFF identifies a code position that is permanently reserved. U+0025 identifies a code position to which a character is allocated; U+0025 also identifies that character (named PERCENT SIGN).

NOTE 2 – These short identifiers are independent of the language in which this standard is written, and are thus retained in all translations of the text.

The following alternative forms of notation of a short identifier are defined here.

- a. The eight-digit form of short identifier shall consist of the sequence of eight hexadecimal digits that represents the code position of the character (see 6.2).
- b. The four-to-six-digit form of short identifier shall consist of the last four to six digits of the eight-digit form. It is not defined if the eight-digit form is greater than 0010FFFF. Leading zeroes beyond four digits are suppressed.
- c. The character “-” (HYPHEN-MINUS) may, as an option, precede the 8-digit form of short identifier.
- d. The character “+” (PLUS SIGN) may, as an option, precede the four-to-six-digit form of short identifier.
- e. The prefix letter “U” (LATIN CAPITAL LETTER U) may, as an option, precede any of the four forms of short identifier defined in a. to d. above.
- f. For the 8 digit forms, the characters SPACE or NO-BREAK SPACE may optionally be inserted before the four last digits.

The capital letters A to F, and U that appear within short identifiers may be replaced by the corresponding small letters.

The full syntax of the notation of a short identifier, in Backus-Naur form, is:

{ U | u } [{+}(xxxx | xxxxx | xxxxxx) | {-}xxxxxxx]

where “x” represents one hexadecimal digit (0 to 9, A to F, or a to f), for example:

-hhhhhhhh +kkkk
Uhhhhhhhh U+kkkk

where hhhhhhhh indicates the eight-digit form and kkkk indicates the four-to-six-digit form.

NOTE 3 – As an example the short identifier for LATIN SMALL LETTER LONG S (see tables for Row 01 in clause 26) may be notated in any of the following forms:

0000017F -0000017F U0000017F U-0000017F
017F +017F U017F U+017F

Any of the capital letters may be replaced by the corresponding small letter.

NOTE 4 – Two special prefixed forms of notation have also been used, in which the letter T (LATIN CAPITAL LETTER T or LATIN SMALL LETTER T) replaces the letter U in the corresponding prefixed forms. The forms of notation that included the prefix letter T indicated that the short identifier refers to a character in ISO/IEC 10646-1 First Edition (before the application of any Amendments), whereas the forms of notation that include the prefix letter U always indicate that the short identifier refers to a character in ISO/IEC 10646 at the most recent state of amendment. Corresponding short identifiers of the form T-xxxxxxx and U-xxxxxxx refer to the same character except when xxxxxxxx lies in the range 00003400 to 00004DFF inclusive. Forms of notation that include no prefix letter always indicate a reference to the most recent state of amendment of ISO/IEC 10646, unless otherwise qualified.

6.6 UCS Sequence Identifiers

ISO/IEC 10646 defines an identifier for any sequence of code positions taken from the standard. Such an identifier is known as a UCS Sequence Identifier (USI). For a sequence of *n* code positions it has the following form:

<UID1, UID2, ..., UID_{*n*}>

where UID1, UID2, etc. represent the short identifiers of the corresponding code positions, in the same order as those code positions appear in the sequence. If each of the code positions in such a sequence has a character allocated to it, the USI can be used to identify the sequence of characters allocated at those code positions. The syntax for UID1, UID2, etc. is specified in clause 6.5. A COMMA character (optionally followed by a SPACE character) separates the UIDs. The UCS Sequence Identifier shall include at least two UIDs; it shall begin with a LESS-THAN SIGN and be terminated by a GREATER-THAN SIGN.

NOTE – UCS Sequences Identifiers cannot be used for specification of subset and collection content. They may be used outside this standard to identify: composite sequences for mapping purposes, font repertoire, etc.

7 General requirements for the UCS

The following requirements apply to the entire coded character set.

- The values of P-, and R-, and C-octets used for representing graphic characters shall be in the

range 00 to FF. The values of G-octets used for representation of graphic characters shall be in the range 00 to 7F. On any plane, code positions FFFE and FFFF shall not be used.

- Code positions to which a character is not allocated, except for the positions reserved for private use characters or for transformation formats, are reserved for future standardization and shall not be used for any other purpose. Future editions of ISO/IEC 10646 will not allocate any characters to code positions reserved for private use characters or for transformation formats.
- The same graphic character shall not be allocated to more than one code position. There are graphic characters with similar shapes in the coded character set; they are used for different purposes and have different character names.

8 The Basic Multilingual Plane

Plane 00 of Group 00 shall be the Basic Multilingual Plane (BMP). The BMP can be used as a two-octet coded character set in which case it shall be called UCS-2 (see 13.1).

NOTE 1 – Since UCS-2 only contains the repertoire of the BMP it is not fully interoperable with UCS-4, UTF-8 and UTF-16.

Code positions 0000 0000 to 0000 001F in the BMP are reserved for control characters, and code position 0000 007F is reserved for the character DELETE (see clause 15). Code positions 0000 0080 to 0000 009F are reserved for control characters.

Code positions 0000 2060 to 0000 206F, 0000 FFF0 to 0000 FFFC, and 000E 0000 to 000E 0FFF are reserved for Alternate Format Characters (see annex F).

NOTE 2 – Unassigned code positions in those ranges may be ignored in normal processing and display.

Code positions 0000 D800 to 0000 DFFF are reserved for the use of UTF-16 (see annex C). These positions are known as the S-zone.

Code positions 0000 E000 to 0000 F8FF are reserved for private use (see clause 10). These positions are known as the private use zone.

Code positions 0000 FDD0 to 0000 FDEF, 0000 FFFE, and 0000 FFFF are permanently reserved.

NOTE 3 – Code position 0000 FFFE is reserved for “signature” (see annex H). Code positions 0000 FDD0 to 0000 FDEF, and 0000 FFFF can be used for internal processing uses requiring numeric values which are guaranteed not to be coded characters, such as in terminating tables, or signaling end-of-text. Furthermore, since 0000 FFFF is the largest BMP value, it may also be used as the final value in binary or sequential searching index within the context of UCS-2 or UTF-16.

NOTE 4 – A “permanently reserved” code position cannot be changed by future amendments.

9 Supplementary planes

9.1 Planes accessible by UTF-16

Each code position in Planes 01 to 10 of Group 00 has a unique mapping to a four-octet sequence in accordance with the UTF-16 form of coded representation (see annex C). This form is compatible with the two-octet BMP form of UCS-2 (see 13.1).

The planes 01, 02 and 0E of Group 00 shall be the Supplementary Multilingual Plane (SMP), the Supplementary Ideographic Plane (SIP) and the Supplementary Special-purpose Plane (SSP) respectively. Like the BMP, these planes contain graphic characters allocated to code positions. The Planes from 03 to 0D of Group 00 are reserved for future standardization. See clause 10.2 for the definition of Plane 0F and 10 of Group 00.

NOTE - The following table shows the boundary code positions for planes 01, 02 and 0E expressed in UCS-4 abbreviated five-digit values and in UTF-16 pairs values.

Plane	UCS-4 values	UTF-16 pairs values
01	10000 - 1FFFF	D800 DC00 - D83F DFFF
02	20000 - 2FFFF	D840 DC00 - D87F DFFF
0E	E0000 - EFFFF	DB40 DC00 - DB7F DFFF

In the UCS Transformation Format UTF-8 (see annex D), the UCS-4 representation of characters shall be used as the source for the mapping. Using the high-half zone value and low-half zone values as source for the mapping is undefined.

NOTE - The following table shows the boundary code positions for planes 01, 02 and 0E expressed in UCS-4 five-digit abbreviated values and in UTF-8 sequence values.

Plane	UCS-4 values	UTF-8 sequence values
01	10000 - 1FFFF	F0908080 - F09FBFBF
02	20000 - 2FFFF	F0A08080 - F0AFBFBF
0E	E0000 - EFFFF	F3A08080 - F3AFBFBF

UCS-2 cannot be used to represent any characters on the Supplementary Planes.

Code positions 1FFFE, 1FFFF, 2FFFE, 2FFFF, EFFFE and EFFFF are permanently reserved.

NOTE – These code positions can be used for internal processing uses requiring a numeric value that is guaranteed not to be a coded character.

9.2 Other Planes reserved for future standardization

Planes 11 to FF in Group 00 and all planes in any other groups (i.e. Planes 00 to FF in Groups 01 to 7F) are reserved for future standardization, and thus those code positions shall not be used for any other purpose.

Code positions in these planes do not have a mapping to the UTF-16 form (see Annex C).

NOTE – To ensure continued interoperability between the UTF-16 form and other coded representations of the UCS, it is intended that no characters will be allocated to code positions in Planes 11 to FF in Group 00 or any planes in any other groups.

10 Private use planes and zones

10.1 Private use characters

Private use characters are not constrained in any way by ISO/IEC 10646. Private use characters can be used to provide user-defined characters. For example, this is a common requirement for users of ideographic scripts.

NOTE 1 – For meaningful interchange of private use characters, an agreement, independent of ISO/IEC 10646, is necessary between sender and recipient.

Private use characters can be used for dynamically-redefinable character applications.

NOTE 2 – For meaningful interchange of dynamically-redefinable characters, an agreement, independent of ISO/IEC 10646 is necessary between sender and recipient. ISO/IEC 10646 does not specify the techniques for defining or setting up dynamically-redefinable characters.

10.2 Code positions for private use characters

The code positions of Plane 0F and Plane 10 of Group 00 shall be for private use.

The 6400 code positions E000 to F8FF of the Basic Multilingual Plane shall be for private use.

The contents of these code positions are not specified in ISO/IEC 10646 (see 10.1).

11 Revision and updating of the UCS

The revision and updating of this coded character set will be carried out by ISO/IEC JTC1/SC2.

NOTE – It is intended that in future editions of ISO/IEC 10646, the names and allocation of the characters in this edition will remain unchanged.

12 Subsets

ISO/IEC 10646 provides the specification of subsets of coded graphic characters for use in interchange, by originating devices, and by receiving devices.

There are two alternatives for the specification of subsets: limited subset and selected subset. An adopted subset may comprise either of them, or a combination of the two.

12.1 Limited subset

A limited subset consists of a list of graphic characters in the specified subset. This specification allows applications and devices that were developed using other codes to interwork with this coded character set.

A claim of conformance referring to a limited subset shall list the graphic characters in the subset by the names of graphic characters or code positions as defined in ISO/IEC 10646.

12.2 Selected subset

A selected subset consists of a list of collections of graphic characters as defined in ISO/IEC 10646. The collections from which the selection may be made are listed in annex A. A selected subset shall always automatically include the Cells 20 to 7E of Row 00 of Plane 00 of Group 00.

A claim of conformance referring to a selected subset shall list the collections chosen as defined in ISO/IEC 10646.

13 Coded representation forms of the UCS

ISO/IEC 10646 provides four alternative forms of coded representation of characters. Two of these forms are specified in this clause, and two others, UTF-16 and UTF-8, are specified in annexes C and D respectively.

NOTE – The characters from the ISO/IEC 646 IRV repertoire are coded by simple zero extensions to their coded representations in ISO/IEC 646 IRV. Therefore, their coded representations have the same integer values when represented as 8-bit, 16-bit, or 32-bit integers. For implementations sensitive to a zero-valued octet (e.g. for use as a string terminator), use of 8-bit based array data type should be avoided as any zero-valued octet may be interpreted incorrectly. Use of data types at least 16-bits wide is more suitable for UCS-2, and use of data types at least 32-bits wide is more suitable for UCS-4.

13.1 Two-octet BMP form

This coded representation form permits the use of characters from the Basic Multilingual Plane with each character represented by two octets.

Within a CC-data-element conforming to the two-octet BMP form, a character from the Basic Multilingual Plane shall be represented by two octets comprising the R-octet and the C-octet as specified in 6.2 (i.e. its RC-element).

NOTE – A coded graphic character using the two-octet BMP form may be implemented by a 16-bit integer for processing.

13.2 Four-octet canonical form (UCS-4)

The canonical form permits the use of all the characters of ISO/IEC 10646, with each character represented by four octets.

Within a CC-data-element conforming to the four-octet canonical form, every character shall be represented by four octets comprising the G-octet, the P-octet, the R-octet, and the C-octet as specified in 6.2.

NOTE 1 – A coded graphic character using the four-octet canonical form may be implemented by a 32-bit integer for processing.

NOTE 2 – When confined to the code positions in Planes 0 to 10 (U+0000 to U+10FFFF), UCS-4 is also referred to as UCS Transformation Format 32 (UTF-32). The Unicode Standard, Version 3.2, defines the following forms of UTF-32:

- UTF-32: the ordering of octets (specified in sub-clause 6.3) is not defined and the signatures (specified in Annex H) may appear;
- UTF-32BE: in the ordering of octets the more significant octets precede the less significant octets, as specified in sub-clause 6.2, and no signatures appear;
- UTF-32LE: in the ordering of octets the less significant octets precede the more significant octets, and no signatures appear.

14 Implementation levels

ISO/IEC 10646 specifies three levels of implementation. Combining characters are described in 24 and listed in annex B.

14.1 Implementation level 1

When implementation level 1 is used, a CC-data-element shall not contain coded representations of combining characters (see clause B.1) nor of characters from the HANGUL JAMO block (see clause 26.1). When implementation level 1 is used the unique-spelling rule shall apply (26.2).

14.2 Implementation level 2

When implementation level 2 is used, a CC-data-element shall not contain coded representations of characters listed in clause B.2. When implementation level 2 is used the unique-spelling rule shall apply (26.2).

14.3 Implementation level 3

When implementation level 3 is used, a CC-data-element may contain coded representations of any characters.

15 Use of control functions with the UCS

This coded character set provides for use of control functions encoded according to ISO/IEC 6429 or similarly structured standards for control functions, and standards derived from these. A set or subset of such coded control functions may be used in conjunction with this coded character set. These standards encode a control function as a sequence of one or more octets.

When a control character of ISO/IEC 6429 is used with this coded character set, its coded representation as specified in ISO/IEC 6429 shall be padded to correspond with the number of octets in the adopted form (see clause 13 and annexes C and D). Thus, the least

significant octet shall be the bit combination specified in ISO/IEC 6429, and the more significant octet(s) shall be zeros.

For example, the control character FORM FEED is represented by "000C" in the two-octet form, and "0000 000C" in the four-octet form.

For escape sequences, control sequences, and control strings (see ISO/IEC 6429) consisting of a coded control character followed by additional bit combinations in the range 20 to 7F, each bit combination shall be padded by octet(s) with value 00.

For example, the escape sequence "ESC 02/00 04/00" is represented by "001B 0020 0040" in the two-octet form, and "0000 001B 0000 0020 0000 0040" in the four-octet form.

NOTE – The term "character" appears in the definition of many of the control functions specified in ISO/IEC 6429, to identify the elements on which the control functions will act. When such control functions are applied to coded characters according to ISO/IEC 10646 the action of those control functions will depend on the type of element from ISO/IEC 10646 that has been chosen, by the application, to be the element (or character) on which the control functions act. These elements may be chosen to be characters (non-combining characters and/or combining characters) or may be chosen in other ways (such as composite sequences) when applicable.

Code extension control functions for the ISO/IEC 2022 code extension techniques (such as designation escape sequences, single shift, and locking shift) shall not be used with this coded character set.

16 Declaration of identification of features

16.1 Purpose and context of identification

CC-data-elements conforming to ISO/IEC 10646 are intended to form all or part of a composite unit of coded information that is interchanged between an originator and a recipient. The identification of ISO/IEC 10646 (including the form), the implementation level, and any subset of the coding space that have been adopted by the originator must also be available to the recipient. The route by which such identification is communicated to the recipient is outside the scope of ISO/IEC 10646.

However, some standards for interchange of coded information may permit, or require, that the coded representation of the identification applicable to the CC-data-element forms a part of the interchanged information. This clause specifies a coded representation for the identification of UCS with an implementation level and a subset of ISO/IEC 10646, and also of a C0 and a C1 set of control functions from ISO/IEC 6429 for use in conjunction with ISO/IEC 10646. Such coded representations provide all or part of an identification data element, which may be included in information interchange in accordance with the relevant standard.

If two or more of the identifications are present, the order of those identifications shall follow the order as specified in this clause.

NOTE – An alternative method of identification is described in annex N.

16.2 Identification of UCS coded representation form with implementation level

When the escape sequences from ISO/IEC 2022 are used, the identification of a coded representation form of UCS (see clause 13) and an implementation level (see clause 14) specified by ISO/IEC 10646 shall be by a designation sequence chosen from the following list:

ESC 02/05 02/15 04/00
UCS-2 with implementation level 1

ESC 02/05 02/15 04/01
UCS-4 with implementation level 1

ESC 02/05 02/15 04/03
UCS-2 with implementation level 2

ESC 02/05 02/15 04/04
UCS-4 with implementation level 2

ESC 02/05 02/15 04/05
UCS-2 with implementation level 3

ESC 02/05 02/15 04/06
UCS-4 with implementation level 3

or from the lists in C.5 and D.6.

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 15.

16.3 Identification of subsets of graphic characters

When the control sequences of ISO/IEC 6429 are used, the identification of subsets (see clause 12) specified by ISO/IEC 10646 shall be by a control sequence IDENTIFY UNIVERSAL CHARACTER SUBSET (IUCS) as shown below.

CSI Ps... 02/00 06/13

Ps... means that there can be any number of selective parameters. The parameters are to be taken from the subset collection numbers as shown in annex A of ISO/IEC 10646. When there is more than one parameter, each parameter value is separated by an octet with value 03/11.

Parameter values are represented by digits where octet values 03/00 to 03/09 represent digits 0 to 9.

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 a control sequence appears within a CC-data-element conforming to ISO/IEC 10646, it shall be padded in accordance with clause 15.

16.4 Identification of control function set

When the escape sequences from ISO/IEC 2022 are used, the identification of each set of control functions (see clause 15) of ISO/IEC 6429 to be used in conjunction with ISO/IEC 10646 shall be an identifier sequence of the type shown below.

ESC 02/01 04/00	identifies the full C0 set of ISO/IEC 6429
ESC 02/02 04/03	identifies the full C1 set of ISO/IEC 6429

For a subset of C0 or C1 sets, the final octet F shall be obtained from the International Register of Coded Character Sets. The identifier sequences for these sets shall be:

ESC 02/01 F	identifies a C0 set
ESC 02/02 F	identifies a C1 set

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 15.

16.5 Identification of the coding system of ISO/IEC 2022

When the escape sequences from ISO/IEC 2022 are used, the identification of a return, or transfer, from UCS to the coding system of ISO/IEC 2022 shall be by the escape sequence ESC 02/05 04/00. If such an escape sequence appears within a CC-data-element conforming to ISO/IEC 10646, it shall be padded in accordance with clause 15.

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

NOTE – Escape sequence ESC 02/05 04/00 is normally used for return to the restored state of ISO/IEC 2022. The escape sequence ESC 02/05 04/00 specified here is sometimes not exactly as specified in ISO/IEC 2022 due to the presence of padding octets. For this reason the escape sequences in 16.2 for the identification of UCS include the octet 02/15 to indicate that the return does not always conform to that standard.

17 Structure of the code tables and lists

The clause 33 sets out the detailed code tables and the lists of character names for the graphic characters. It specifies graphic characters, their coded representation, and the character name for each character.

The graphic symbols are to be regarded as typical visual representations of the characters. ISO/IEC 10646 does not attempt to prescribe the exact shape of each character. The shape is affected by the design of the font employed, which is outside the scope of ISO/IEC 10646.

Graphic characters specified in ISO/IEC 10646 are uniquely identified by their names. This does not imply that the graphic symbols by which they are commonly imaged are always different. Examples of graphic characters with similar graphic symbols are LATIN CAPITAL LETTER A, GREEK CAPITAL LETTER ALPHA and CYRILLIC CAPITAL LETTER A.

The meaning attributed to any character is not specified by ISO/IEC 10646; it may differ from country to country, or from one application to another.

For the alphabetic scripts, the general principle has been to arrange the characters within any row in approximate alphabetic sequence; where the script has capital and small letters, these are arranged in pairs. However, this general principle has been overridden in some cases. For example, for those scripts for which a relevant standard exists, the characters are allocated according to that standard. This arrangement within the code tables will aid conversion between the existing standards and this coded character set. In general, however, it is anticipated that conversion between this coded character set and any other coded character set will use a table lookup technique.

It is not intended, nor will it often be the case, that the characters needed by any one user will be found all grouped together in one part of the code table.

Furthermore, the user of any script will find that needed characters may have been coded elsewhere in this coded character set. This especially applies to the digits, to the symbols, and to the use of Latin letters in dual-script applications.

Therefore, in using this coded character set, the reader is advised to refer first to the block names list in annex A.2 or an overview of the Planes in figures 3 to 7, and then to turn to the specific code table rows for the relevant script and for symbols and digits. In addition, annex G contains an alphabetically sorted list of character names.

18 Block names

Named blocks of contiguous code positions are specified within a plane for the purpose of allocation of

characters sharing some common characteristic, such as script. The blocks specified within the BMP, SMP, SIP and SSP are listed in A.2 of annex A, and are illustrated in figures 3 to 7.

19 Characters in bidirectional context

A class of left/right handed pairs of characters has special significance in the context of bidirectional text. In this context the terms LEFT or RIGHT in the character name are also intended to imply “opening” or “closing” forms of character shape, rather than a strict left-hand or right-hand form. These characters are listed below.

<u>Code</u> <u>Position</u>	<u>Name</u>
0028	LEFT PARENTHESIS
0029	RIGHT PARENTHESIS
005B	LEFT SQUARE BRACKET
005D	RIGHT SQUARE BRACKET
007B	LEFT CURLY BRACKET
007D	RIGHT CURLY BRACKET
2045	LEFT SQUARE BRACKET WITH QUILL
2046	RIGHT SQUARE BRACKET WITH QUILL
207D	SUPERSCRIPIT LEFT PARENTHESIS
207E	SUPERSCRIPIT RIGHT PARENTHESIS
208D	SUBSCRIPT LEFT PARENTHESIS
208E	SUBSCRIPT RIGHT PARENTHESIS
2329	LEFT-POINTING ANGLE BRACKET
232A	RIGHT-POINTING ANGLE BRACKET
3008	LEFT ANGLE BRACKET
3009	RIGHT ANGLE BRACKET
300A	LEFT DOUBLE ANGLE BRACKET
300B	RIGHT DOUBLE ANGLE BRACKET
300C	LEFT CORNER BRACKET
300D	RIGHT CORNER BRACKET
300E	LEFT WHITE CORNER BRACKET
300F	RIGHT WHITE CORNER BRACKET
3010	LEFT BLACK LENTICULAR BRACKET
3011	RIGHT BLACK LENTICULAR BRACKET
3014	LEFT TORTOISE SHELL BRACKET
3015	RIGHT TORTOISE SHELL BRACKET
3016	LEFT WHITE LENTICULAR BRACKET
3017	RIGHT WHITE LENTICULAR BRACKET
3018	LEFT WHITE TORTOISE SHELL BRACKET
3019	RIGHT WHITE TORTOISE SHELL BRACKET
301A	LEFT WHITE SQUARE BRACKET
301B	RIGHT WHITE SQUARE BRACKET

The interpretation and rendering of any of these characters depend on the state related to the symmetric swapping characters (see F.2.2) and on the direction of the character being rendered that are in effect at the point in the CC-data-element where the coded representation of the character appears.

For example, if the character ACTIVATE SYMMETRIC SWAPPING occurs and if the direction of the character is from right to left, the character shall be interpreted as if the term LEFT or RIGHT in its name had

been replaced by the term RIGHT or LEFT, respectively.

NOTE – In the context of Arabic bidirectional text, certain mathematical symbols may also have special significance (see annex E).

19.1 Directionality of bidirectional text

The Unicode Bidirectional Algorithm describes the algorithm used to determine the directionality for bidirectional text.

20 Special characters

There are some characters that do not have printable graphic symbols or are otherwise special in some ways.

20.1 Space characters

The following characters are space characters. They are

<u>Code</u> <u>Position</u>	<u>Name</u>
0020	SPACE
00A0	NO-BREAK SPACE
2000	EN QUAD
2001	EM QUAD
2002	EN SPACE
2003	EM SPACE
2004	THREE-PER-EM SPACE
2005	FOUR-PER-EM SPACE
2006	SIX-PER-EM SPACE
2007	FIGURE SPACE
2008	PUNCTUATION SPACE
2009	THIN SPACE
200A	HAIR SPACE
3000	IDEOGRAPHIC SPACE

20.2 Currency symbols

Currency symbols in ISO/IEC 10646 do not necessarily identify the currency of a country. For example, YEN SIGN can be used for Japanese Yen and Chinese Yuan. Also, DOLLAR SIGN is used in numerous countries including the United States of America.

20.3 Alternate Format Characters

There is a special class of characters called Alternate Format Characters which are included for compatibility with some industry practices. They are:

00AD	SOFT HYPHEN
058A	ARMENIAN HYPHEN
1806	MONGOLIAN TODO SOFT HYPHEN
180E	MONGOLIAN VOWEL SEPARATOR
200B	ZERO WIDTH SPACE
200C	ZERO WIDTH NON-JOINER
200D	ZERO WIDTH JOINER
200E	LEFT-TO-RIGHT MARK
200F	RIGHT-TO-LEFT MARK
2028	LINE SEPARATOR
2029	PARAGRAPH SEPARATOR
202A	LEFT-TO-RIGHT EMBEDDING

202B	RIGHT-TO-LEFT EMBEDDING
202C	POP DIRECTIONAL FORMATTING
202D	LEFT-TO-RIGHT OVERRIDE
202E	RIGHT-TO-LEFT OVERRIDE
202F	NARROW NO-BREAK SPACE
206A	INHIBIT SYMMETRIC SWAPPING
206B	ACTIVATE SYMMETRIC SWAPPING
206C	INHIBIT ARABIC FORM SHAPING
206D	ACTIVATE ARABIC FORM SHAPING
206E	NATIONAL DIGIT SHAPES
206F	NOMINAL DIGIT SHAPES
2FF0	IDEOGRAPHIC DESCRIPTION CHARACTER LEFT TO RIGHT
2FF1	IDEOGRAPHIC DESCRIPTION CHARACTER ABOVE TO BELOW
2FF2	IDEOGRAPHIC DESCRIPTION CHARACTER LEFT TO MIDDLE AND RIGHT
2FF3	IDEOGRAPHIC DESCRIPTION CHARACTER ABOVE TO MIDDLE AND BELOW
2FF4	IDEOGRAPHIC DESCRIPTION CHARACTER FULL SURROUND
2FF5	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM ABOVE
2FF6	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM BELOW
2FF7	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM LEFT
2FF8	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM UPPER LEFT
2FF9	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM UPPER RIGHT
2FFA	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM LOWER LEFT
2FFB	IDEOGRAPHIC DESCRIPTION CHARACTER OVERLAID
	3164 HANGUL FILLER
FEFF	ZERO WIDTH NO-BREAK SPACE
FFA0	HALFWIDTH HANGUL FILLER
FFF9	INTERLINEAR ANNOTATION ANCHOR
FFFA	INTERLINEAR ANNOTATION SEPARATOR
FFFB	INTERLINEAR ANNOTATION TERMINATOR

These characters are described in annex F.

20.4 Variation selectors

Variation selectors are combining characters following immediately a specific base character to indicate a specific variant form of graphic symbol for that character. Some variation selectors are specific to a script, such as the Mongolian free variation selectors, others are used with various other base characters such as the mathematical symbols. Variations selectors following other characters have no effect on the selection of the graphic symbol for that character.

No sequences using characters from VARIATION SELECTOR-2 to VARIATION SELECTOR-16 from the Basic Multilingual Plane and VARIATION SELECTOR-17 to VARIATION SELECTOR-256 from the Supplementary Special-purpose Plane are defined at this time.

The following table provides a description of the variant appearances corresponding to the use of appropriate variation selectors with all allowed base mathematical symbols.

NOTE 1 – The VARIATION SELECTOR-1 is the only variation selector used with mathematical symbols.

Sequence (UID notation)	Description of variant appearance
<2229, FE00>	INTERSECTION with serifs
<222A, FE00>	UNION with serifs
<2268, FE00>	LESS-THAN BUT NOT EQUAL TO with vertical stroke
<2269, FE00>	GREATER-THAN BUT NOT EQUAL TO with vertical stroke
<2272, FE00>	LESS-THAN OR EQUIVALENT TO following the slant of the lower leg
<2273, FE00>	GREATER-THAN OR EQUIVALENT TO following the slant of the lower leg
<228A, FE00>	SUBSET OF WITH NOT EQUAL TO with stroke through bottom members
<228B, FE00>	SUPERSET OF WITH NOT EQUAL TO with stroke through bottom members
<2293, FE00>	SQUARE CAP with serifs
<2294, FE00>	SQUARE CUP with serifs
<2295, FE00>	CIRCLED PLUS with white rim
<2297, FE00>	CIRCLED TIMES with white rim
<229C, FE00>	CIRCLED EQUALS equal sign touching the circle
<22DA, FE00>	LESS-THAN EQUAL TO OR GREATER-THAN with slanted equal
<22DB, FE00>	GREATER-THAN EQUAL TO OR LESS-THAN with slanted equal
<2A3C, FE00>	INTERIOR PRODUCT tall variant with narrow foot
<2A3D, FE00>	RIGHTHAND INTERIOR PRODUCT tall variant with narrow foot
<2A9D, FE00>	SIMILAR following the slant of the upper leg OR LESS-THAN
<2A9E, FE00>	SIMILAR following the slant of the upper leg OR GREATER-THAN
<2AAC, FE00>	SMALLER THAN OR EQUAL TO with slanted equal
<2AAD, FE00>	LARGER THAN OR EQUAL TO with slanted equal

<2ACB, FE00>	SUBSET OF ABOVE NOT EQUAL TO with stroke through bottom members
<2ACC, FE00>	SUPERSET OF ABOVE NOT EQUAL TO with stroke through bottom members

The following table provides a description of the variant appearances corresponding to the use of appropriate variation selectors with all allowed base Mongolian characters. Only some presentation forms of the base Mongolian characters used with the Mongolian free variation selectors produce variant appearances. These combinations are described in the following table.

NOTE 2 – The Mongolian characters have various presentation forms depending on their position in a CC-data element. These presentations forms are called isolate, initial, medial and final.

<u>Sequence (UID notation)</u>	<u>position</u>	<u>Description of variant appearance</u>
<1820, 180B>	isolate, medial, final	MONGOLIAN LETTER A second form
<1820, 180C>	medial	MONGOLIAN LETTER A third form
<1821, 180B>	initial, final	MONGOLIAN LETTER E second form
<1822, 180B>	medial	MONGOLIAN LETTER I second form
<1823, 180B>	medial, final	MONGOLIAN LETTER O second form
<1824, 180B>	medial	MONGOLIAN LETTER U second form
<1825, 180B>	medial, final	MONGOLIAN LETTER OE second form
<1825, 180C>	medial	MONGOLIAN LETTER OE third form
<1826, 180B>	isolate, medial, final	MONGOLIAN LETTER UE second form
<1826, 180C>	medial	MONGOLIAN LETTER UE third form
<1828, 180B>	initial, medial	MONGOLIAN LETTER NA second form
<1828, 180C>	medial	MONGOLIAN LETTER NA third form
<1828, 180D>	medial	MONGOLIAN LETTER NA separate form
<182A, 180B>	final	MONGOLIAN LETTER BA alternative form
<182C, 180B>	initial, medial	MONGOLIAN LETTER QA second form

<182C, 180B>	isolate	MONGOLIAN LETTER QA feminine second form
<182C, 180C>	medial	MONGOLIAN LETTER QA third form
<182C, 180D>	medial	MONGOLIAN LETTER QA fourth form
<182D, 180B>	initial, medial	MONGOLIAN LETTER GA second form
<182D, 180B>	final	MONGOLIAN LETTER GA feminine form
<182D, 180C>	medial	MONGOLIAN LETTER GA third form
<182D, 180D>	medial	MONGOLIAN LETTER GA feminine form
<1830, 180B>	final	MONGOLIAN LETTER SA second form
<1830, 180C>	final	MONGOLIAN LETTER SA third form
<1832, 180B>	medial	MONGOLIAN LETTER TA second form
<1833, 180B>	initial, medial, final	MONGOLIAN LETTER DA second form
<1835, 180B>	final	MONGOLIAN LETTER JA second form
<1836, 180B>	initial, medial	MONGOLIAN LETTER YA second form
<1836, 180C>	medial	MONGOLIAN LETTER YA third form
<1838, 180B>	final	MONGOLIAN LETTER WA second form
<1844, 180B>	medial	MONGOLIAN LETTER TODO E second form
<1845, 180B>	medial	MONGOLIAN LETTER TODO I second form
<1846, 180B>	medial	MONGOLIAN LETTER TODO O second form
<1847, 180B>	isolate, medial, final	MONGOLIAN LETTER TODO U second form
<1847, 180C>	medial	MONGOLIAN LETTER TODO U third form
<1848, 180B>	medial	MONGOLIAN LETTER TODO OE second form
<1849, 180B>	isolate, medial	MONGOLIAN LETTER TODO UE second form
<184D, 180B>	initial, medial	MONGOLIAN LETTER TODO QA feminine form
<184E, 180B>	medial	MONGOLIAN LETTER TODO GA second form
<185D, 180B>	medial, final	MONGOLIAN LETTER SIBE E second form

<185E, 180B>	medial, final	MONGOLIAN LETTER SIBE I second form
<185E, 180C>	medial, final	MONGOLIAN LETTER SIBE I third form
<1860, 180B>	medial, final	MONGOLIAN LETTER SIBE UE second form
<1863, 180B>	medial	MONGOLIAN LETTER SIBE KA second form
<1868, 180B>	initial, medial	MONGOLIAN LETTER SIBE TA second form
<1868, 180C>	medial	MONGOLIAN LETTER SIBE TA third form
<1869, 180B>	initial, medial	MONGOLIAN LETTER SIBE DA second form
<186F, 180B>	initial, medial	MONGOLIAN LETTER SIBE ZA second form
<1873, 180B>	medial, final	MONGOLIAN LETTER MANCHU I second form
<1873, 180C>	medial, final	MONGOLIAN LETTER MANCHU I third form
<1873, 180D>	medial	MONGOLIAN LETTER MANCHU I fourth form
<1874, 180B>	medial	MONGOLIAN LETTER MANCHU KA second form
<1874, 180B>	final	MONGOLIAN LETTER MANCHU KA feminine first form
<1874, 180C>	medial	MONGOLIAN LETTER MANCHU KA feminine first form
<1874, 180C>	final	MONGOLIAN LETTER MANCHU KA feminine sec- ond form
<1874, 180D>	medial	MONGOLIAN LETTER MANCHU KA feminine sec- ond form
<1876, 180B>	initial, medial	MONGOLIAN LETTER MANCHU FA second form
<1880, 180B>	all	MONGOLIAN LETTER ALI GALI ANUSVARA ONE sec- ond form
<1881, 180B>	all	MONGOLIAN LETTER ALI GALI VISARGA ONE sec- ond form
<1887, 180B>	isolate, final	MONGOLIAN LETTER ALI GALI A second form
<1887, 180C>	final	MONGOLIAN LETTER ALI GALI A third form
<1887, 180D>	final	MONGOLIAN LETTER ALI GALI A fourth form
<1888, 180B>	final	MONGOLIAN LETTER ALI GALI I second form
<188A, 180B>	initial, medial	MONGOLIAN LETTER ALI GALI NGA second form

NOTE 3 – The variation selector only selects a different *appearance* of an already encoded character. It is not intended as a general code extension mechanism. Only the sequences specifically defined in this clause are sanctioned for standard use; all other sequences are undefined. No sequences containing combining characters or composite characters will be defined.

NOTE 4 – The exhaustive list of standardized variants is also described as *StandardizedVariants.html* in the Unicode character database.

20.5 Format characters for musical symbols

The following characters are format characters used for the presentation of musical symbols.

1D159	MUSICAL SYMBOL NULL NOTEHEAD
1D173	MUSICAL SYMBOL BEGIN BEAM
1D174	MUSICAL SYMBOL END BEAM
1D175	MUSICAL SYMBOL BEGIN TIE
1D176	MUSICAL SYMBOL END TIE
1D177	MUSICAL SYMBOL BEGIN SLUR
1D178	MUSICAL SYMBOL END SLUR
1D179	MUSICAL SYMBOL BEGIN PHRASE
1D17A	MUSICAL SYMBOL END PHRASE

These characters are further described in Annex U.

20.6 Tag characters

The functionality of the TAGS characters, part of the TAGS block within the Supplementary Special-purpose Plane (SSP), is not specified by this international standard.

NOTE - However the intended use of these characters is described in annex T.

21 Presentation forms of characters

Each presentation form of a character provides an alternative form, for use in a particular context, to the nominal form of the character or sequence of characters from the other zones of graphic characters. The transformation from the nominal form to the presentation forms may involve substitution, superimposition, or combination.

The rules for the superimposition, choice of differently shaped characters, or combination into ligatures, or conjuncts, which are often of extreme complexity, are not specified in ISO/IEC 10646.

In general, presentation forms are not intended to be used as a substitute for the nominal forms of the graphic characters specified elsewhere within this coded character set. However, specific applications may encode these presentation forms instead of the nominal forms for specific reasons among which is compatibility with existing devices. The rules for searching, sorting, and other processing operations on presentation forms are outside the scope of ISO/IEC 10646.

Within the BMP these characters are mostly allocated to positions in rows FB to FF.

22 Compatibility characters

Compatibility characters are included in ISO/IEC 10646 primarily for compatibility with existing coded character sets to allow two-way code conversion without loss of information.

Within the BMP many of these characters are allocated to positions within rows F9, FA, FE, and FF, and within rows 31 and 33. Some compatibility characters are also allocated within other rows.

Within the Supplementary Ideographic Plane (SIP) these characters are allocated to positions within rows F8 to FA.

The CJK compatibility ideographs (characters part of the CJK COMPATIBILITY IDEOGRAPHS-2001 collection) are ideographs that should have been unified with one of the CJK unified ideographs (characters part of the CJK UNIFIED IDEOGRAPHS-2001 collection), per the unification rule described in Annex S of this International Standard.

However, they are included in this International Standard as separate characters, because, based on various national, cultural, or historical reasons for some specific country and region, some national and regional standards assign separate code positions for them.

NOTE – For this reason, compatibility ideographs should only be used for maintaining and guaranteeing a round trip conversion with the specific national, regional, or other standard. Other usage is strongly discouraged.

23 Order of characters

Usually, coded characters appear in a CC-data-element in logical order (logical or backing store order corresponds approximately to the order in which characters are entered from the keyboard, after corrections such as insertions, deletions, and overtyping have taken place). This applies even when characters of different dominant direction are mixed: left-to-right (Greek, Latin, Thai) with right-to-left (Arabic, Hebrew), or with vertical (Mongolian) script.

Some characters may not appear linearly in final rendered text. For example, the medial form of DEVANAGARI VOWEL SIGN I is displayed before the character that it logically follows in the CC-data-element.

24 Normalization forms

Normalization forms are the mechanisms allowing the selection of a unique coded representation among

alternative, but equivalent coded text representations of the same text. Normalization forms for use with ISO/IEC 10646 are specified in the Unicode Standard UAX#15.

NOTE 1 – By definition, the result of applying any of these normalization forms is stable over time. It means that a normalized representation of text remains normalized even when the standard is amended.

NOTE 2 – Some normalization forms favor composite sequences over shorter representations of text, others favor the shorter representations. The backward compatibility requirement is provided by establishing ISO/IEC 10646-1:2000 (2nd Edition) and ISO/IEC 10646-2:2001 (1st Edition) as the reference versions for the definition of the shorter representation of text.

25 Combining characters

This clause specifies the use of combining characters. A list of combining characters is shown in clause B.1. A list of combining characters not allowed in implementation level 2 is shown in clause B.2.

NOTE - The names of many script-independent combining characters contain the word “COMBINING”.

25.1 Order of combining characters

Coded representations of combining characters shall follow that of the graphic character with which they are associated (for example, coded representations of LATIN SMALL LETTER A followed by COMBINING TILDE represent a composite sequence for Latin “ã”). If a combining character is to be regarded as a composite sequence in its own right, it shall be coded as a composite sequence by association with the character SPACE. For example, grave accent can be composed as SPACE followed by COMBINING GRAVE ACCENT.

NOTE – Indic matras form a special category of combining characters, since the presentation can depend on more than one of the surrounding characters. Thus it might not be desirable to associate Indic matra with the character SPACE.

25.2 Appearance in code tables

Combining characters intended to be positioned relative to the associated character are depicted within the character code tables above, below, to the right of, to the left of, in, around, or through a dotted circle. In presentation, these characters are intended to be positioned relative to the preceding base character in some manner, and not to stand alone or function as base characters. This is the motivation for the term “combining”. Diacritics are the principal class of combining characters used in European alphabets.

In the code tables for some scripts, such as Hebrew, Arabic, and the scripts of India and South East Asia, combining characters are indicated in relation to dotted circles to show their position relative to the base character. Many of these combining characters en-

code vowel letters; as such they are not generally referred to as “diacritical marks”.

25.3 Alternate coded representations

Alternate coded representations of text are generated by using multiple combining characters in different orders, or using various equivalent combinations of characters and composite sequences. These alternate coded representations result in multiple representations of the same text. Normalizing these coded representations creates a unique representation.

NOTE – For example, in implementation level 3 the French word “là” may be represented by the characters LATIN SMALL LETTER L followed by LATIN SMALL LETTER A WITH GRAVE, or may be represented by the characters LATIN SMALL LETTER L followed by LATIN SMALL LETTER A followed by COMBINING GRAVE ACCENT. When the normalizations forms are applied on those alternate coded representations, only one representation remains. The form of the remaining representation depends on the normalization form used.

25.4 Multiple combining characters

There are instances where more than one combining character is applied to a single graphic character. ISO/IEC 10646 does not restrict the number of combining characters that can follow a base character. The following rules shall apply:

- a. If the combining characters can interact in presentation (for example, COMBINING MACRON and COMBINING DIAERESIS), then the position of the combining characters in the resulting graphic display is determined by the order of the coded representation of the combining characters. The presentations of combining characters are to be positioned from the base character outward. For example, combining characters placed above a base character are stacked vertically, starting with the first encountered in the sequence of coded representations and continuing for as many marks above as are required by the coded combining characters following the coded base character. For combining characters placed below a base character, the situation is inverted, with the combining characters starting from the base character and stacking downward.

An example of multiple combining characters above the base character is found in Thai, where a consonant letter can have above it one of the vowels 0000 0E34 to 0000 0E37 and, above that, one of four tone marks 0000 0E48 to 0000 0E4B. The order of the coded representation is: base consonant, followed by a vowel, followed by a tone mark.

- b. Some specific combining characters override the default stacking behavior by being positioned horizontally rather than stacking, or by forming a ligature with an adjacent combining character.

When positioned horizontally, the order of coded representations is reflected by positioning in the dominant order of the script with which they are used. For example, horizontal accents in a left-to-right script are coded left-to-right.

Prominent characters that show such override behavior are associated with specific scripts or alphabets. For example, the COMBINING GREEK KORONIS (0000 0343) requires that, together with a following acute or grave accent, they be rendered side-by-side above a letter, rather than the accent marks being stacked above the COMBINING GREEK KORONIS. The order of the coded representations is: the letter itself, followed by that of the breathing mark, followed by that of the accent marks. Two Vietnamese tone marks which have the same graphic appearance as the Latin acute and grave accent marks do not stack above the three Vietnamese vowel letters which already contain the circumflex diacritic (â, ê, ô). Instead, they form ligatures with the circumflex component of the vowel letters.

- c. If the combining characters do not interact in presentation (for example, when one combining character is above a graphic character and another is below), the resultant graphic symbol from the base character and combining characters in different orders may appear the same. For example, the coded representations of LATIN SMALL LETTER A, followed by COMBINING CARON, followed by COMBINING OGONEK may result in the same graphic symbol as the coded representations of LATIN SMALL LETTER A, followed by COMBINING OGONEK, followed by COMBINING CARON.

Combining characters in Hebrew or Arabic scripts do not normally interact. Therefore, the sequence of their coded representations in a composite sequence does not affect its graphic symbol. The rules for forming the combined graphic symbol are beyond the scope of ISO/IEC 10646.

25.5 Collections containing combining characters

In some collections of characters listed in annex A, such as collections 14 (BASIC ARABIC) or 25 (THAI), both combining characters and non-combining characters are included.

When implementation level 1 or 2 is adopted, a CC-data-element shall not contain the coded representations of combining characters listed in annex B, even though the adopted subset may include them.

Other collections of characters listed in annex A comprise only combining characters, for example collection 7 (COMBINING DIACRITICAL MARKS). Such a

collection shall not be included in the adopted subset when implementation level 1 is adopted.

26 Special features of individual scripts

26.1 Hangul syllable composition method

In rendering, a sequence of Hangul Jamo (from HANGUL JAMO block: 1100 to 11FF) is displayed as a series of syllable blocks. Jamo can be classified into three classes: Choseong (syllable-initial character), Jungseong (syllable-peak character), and Jongseong (syllable-final character). A complete syllable block is composed of a Choseong and a Jungseong, and optionally a Jongseong.

An incomplete syllable is a string of one or more characters which does not constitute a complete syllable (for example, a Choseong alone, a Jungseong alone, a Jongseong alone, or a Jungseong followed by a Jongseong). An incomplete syllable which starts with a Jungseong or a Jongseong shall be preceded by a CHOSEONG FILLER (0000 115F). An incomplete syllable composed of a Choseong alone shall be followed by a JUNGSEONG FILLER (0000 1160).

The implementation level 3 shall be used for the Hangul syllable composition method.

NOTE 1 – Hangul Jamo are not combining characters.

NOTE 2 – When a combining character such as HANGUL SINGLE DOT TONE MARK (0000 302E) is intended to apply to a sequence of Hangul Jamo it should be placed at the end of the sequence, after the Hangul Jamo character which completes the syllable block.

26.2 Features of Indic alphabetic scripts

In the tables for Rows 09 to 0D and 0F, and for the MYANMAR block in Row 10, of the BMP (see clause 33) the graphic symbols shown for some characters appear to be formed as compounds of the graphic symbols for two other characters in the same table.

Examples:

Row 0B Tamil.

The graphic symbol for 0B94 TAMIL LETTER AU appears as if it is constructed from the graphic symbols for:
0B93 TAMIL LETTER OO and 0BD7 TAMIL AU LENGTH MARK

Row 0D Malayalam.

The graphic symbol for 0D4A MALAYALAM VOWEL SIGN O appears as if it is constructed from the graphic symbols for:
0D46 MALAYALAM VOWEL SIGN E and 0D3E MALAYALAM VOWEL SIGN AA

In such cases a single coded character may appear to the user to be equivalent to the sequence of two

coded characters whose graphic symbols, when combined, are visually similar to the graphic symbol of that single character, as in a composite sequence (4.14).

A “unique-spelling” rule is defined as follows. According to this rule, no coded character from a table for Rows 09 to 0D or 0F, or for the MYANMAR block in Row 10, shall be regarded as equivalent to a sequence of two or more other coded characters taken from the same table.

This “unique-spelling” rule shall apply in Levels 1 and 2.

NOTE – In Levels 1 and 2, if such a sequence occurs in a CC-data-element it is always made available to the user as two distinct characters in accordance with their respective character names.

27 Source references for CJK Ideographs

A CJK Ideograph is always referenced by at least one source reference. These source references are provided in a machine-readable format that is accessible as links to this document. The content pointed by these links is also normative.

NOTE – The referenced files are only available to users who obtain their copy of the standard in a machine-readable format. However, the file format makes them printable.

27.1 Source references for CJK Unified Ideographs

The procedures that were used to derive the unified ideographs from the source character set standards, and the rules for their arrangement in the code tables on the following pages, are described in annex S.

NOTE 1 – The source separation rule described by the clause S.1.6 of that annex only apply to CJK Unified Ideographs within the BMP.

The following list identifies all sources referenced by the CJK Unified Ideographs in both Plane 0 (BMP) and Plane 2 (SIP). The set of CJK Unified Ideographs is represented by the collection CJK UNIFIED IDEOGRAPHS-2001 (See annex A.1).

The Hanzi G sources are

G0	GB2312-80
G1	GB12345-90 with 58 Hong Kong and 92 Korean “Idu” characters
G3	GB7589-87 unsimplified forms
G5	GB7590-87 unsimplified forms
G7	General Purpose Hanzi List for Modern Chinese Language, and General List of Simplified Hanzi
GS	Singapore Characters
G8	GB8565-88
GE	GB16500-95

G_KX	Kangxi Dictionary ideographs (康熙字典) including the addendum (康熙字典) 補遺
G_HZ	Hanyu Dazidian ideographs (漢語大字典)
G_CY	Ci Yuan (辭源)
G_CH	Ci Hai (辭海)
G_HC	Hanyu Dazidian (漢語大詞典)
G_BK	Chinese Encyclopedia (中國大百科全書)
G_FZ	Founder Press System (方正排版系統)
G_4K	Siku Quanshu (四庫全書)

The Hanzi H source is

H Hong Kong Supplementary Character Set

Hanzi T sources are

T1	TCA-CNS 11643-1992 1 st plane
T2	TCA-CNS 11643-1992 2 nd plane
T3	TCA-CNS 11643-1992 3 rd plane with some additional characters
T4	TCA-CNS 11643-1992 4 th plane
T5	TCA-CNS 11643-1992 5 th plane
T6	TCA-CNS 11643-1992 6 th plane
T7	TCA-CNS 11643-1992 7 th plane
TF	TCA-CNS 11643-1992 15 th plane

Kanji J sources are

J0	JIS X 0208-1990
J1	JIS X 0212-1990
J3	JIS X 0213:2000 level-3
J4	JIS X 0213:2000 level-4
JA	Unified Japanese IT Vendors Contemporary Ideographs, 1993

Hanja K sources are

K0	KS C 5601-1987
K1	KS C 5657-1991
K2	PKS C 5700-1 1994
K3	PKS C 5700-2 1994
K4	PKS 5700-3:1998

Hanja KP sources are

KP0	KPS 9566-97
KP1	KPS 10721-2000

ChuNom V sources are

V0	TCVN 5773:1993
V1	TCVN 6056:1995
V2	VHN 01:1998
V3	VHN 02: 1998

The content linked to is a plain text file, using ISO/IEC 646-IRV characters with LINE FEED as end of line mark, that specifies, after a 11-lines header, as many lines as CJK Unified Ideographs in the sum of the two planes; each containing the following information organized in fixed width fields:

- 01-05 octet: Plane 0 or Plane 2 code position (0hhhh), (2hhhh)
- 06-12 octet: Hanzi G sources (G0-hhhh), (G1-hhhh), (G3-hhhh), (G5-hhhh), (G7-hhhh), (G8-hhhh), (G8-hhhh), (GE-hhhh), (G_KX), (G_HZ), (G_CY), (G_CH), (G_HC), (G_BK), (G_FZ) or (G_4K).
- 13-19 octet: Hanzi T sources (T1-hhhh), (T2-hhhh), (T3-hhhh), (T4-hhhh), (T5-hhhh), (T6-hhhh), (T7-hhhh) or (TF-hhhh).
- 20-26 octet: Kanji J sources (J0-hhhh), (J1-hhhh), (J3-hhhh), (J4-hhhh) or (JA-hhhh).
- 27-33 octet: Hanja K source (K0-hhhh), (K1-hhhh), (K2-hhhh), (K3-hhhh) or (K4-dddd).
- 34-40 octet: ChuNom V sources (V0-hhhh), (V1-hhhh), (V2-hhhh) or (V3-hhhh).
- 41-47 octet: Hanzi H source (H-hhhh).
- 48-55 octet: Hanja KP sources (KP0-hhhh) or (KP1-hhhh).

The format definition uses 'd' as a decimal unit and 'h' as a hexadecimal unit. Uppercase characters and all other symbols between parentheses including the space character appear as shown.

[Click on this highlighted text to access the reference file.](#)

NOTE 2 – The content is also available as a separate viewable file in the same file directory as this document. The file is named: "CJKUA_SR.txt".

27.2 Source reference presentation for BMP CJK Unified Ideographs

In the BMP code tables, entries for both CJK Unified Ideographs and its Extension A are arranged as follows.

Row/Cell Hex code	C G- Hanzi -T		J Kanji	K Hanja	V ChuNom
078/000	→	→	→	→	→
4E00	0-523B 0-5027	1-4421 1-3601	0-306C 0-1676	0-6C69 0-7673	1-2121 1-0101

NOTE - Under each ideograph the two lines of numbers indicate the source code positions; the first line shows hexadecimal values, the second line shows decimal values.

The leftmost column of an entry shows the code position in ISO/IEC 10646, giving the code representation both in decimal and in hexadecimal notation.

Each of the other columns shows the graphic symbol for the character, and its coded representation, as specified in a source standard for character sets that is also identified in the table entry. Each of these source standards is assigned to one of five groups indicated by G, T, J, K, or V as shown in the lists below. In each table entry, a separate column is assigned for the corresponding character (if any) from each of those groups of source standards.

An entry in any of the G, T, J, K, or V columns includes a sample graphic symbol from the source character set standard, together with its coded representation in that standard. The first line below the graphic symbol shows the coded representation in hexadecimal notation. The second line shows the coded representation in decimal notation which comprises two digits for section number followed by two digits for position number. Each of the coded representations is prefixed by a one-character source identification followed by a hyphen. This source character identifies the coded character set standard from which the character is taken as shown in the lists above.

27.3 Source references for CJK Compatibility Ideographs

The following list identifies all sources referenced by the CJK Compatibility Ideographs in both Plane 0 (BMP) and Plane 2 (SIP). The set of CJK Compatibility Ideographs is represented by the collection CJK COMPATIBILITY IDEOGRAPHS-2003 (See annex A.1).

The Hanzi H source is:

H Hong Kong Supplementary Character Set

Hanzi T sources are:

T3 TCA-CNS 11643-1992 3rd plane
T4 TCA-CNS 11643-1992 4th plane
T5 TCA-CNS 11643-1992 5th plane
T6 TCA-CNS 11643-1992 6th plane
T7 TCA-CNS 11643-1992 7th plane
TF TCA-CNS 11643-1992 15th plane

Kanji J sources are:

J3 JIS X 213:2000 level-3

J4 JIS X 213:2000 level-4

The Hanja K source is:

K0 KS C 5601-1987

The Hanja KP source is:

KP1 KPS 10721-2000

The Unicode U source is:

U0 The Unicode Standard 3.0-2000

The content linked to is a plain text file, using ISO/IEC 646-IRV characters with LINE FEED as end of line mark, that specifies, after a 11-lines header, as many lines as CJK Compatibility Ideographs; each containing the following information organized in fixed width fields:

- 01-06 octet: Plane 0 or Plane 2 code position (0hhhh) or (2hhhh).
- 07-12 octet: Corresponding CJK Unified Ideograph (0hhhh) or (2hhhh).
- 13-20 octet: Hanzi T sources (T3-hhhh), (T4-hhhh), (T5-hhhh), (T6-hhhh), (T7-hhhh), or (TF-hhhh).
- 21-27 octet: Hanzi H sources (H-hhhh).
- 28-35 octet: Kanji J sources (J3-hhhh), (J4-hhhh).
- 36-43 octet: Hanja K sources (K0-hhhh).
- 44-51 octet: Unicode U sources (U0-hhhh)
- 52-59 octet: Hanja KP sources (KP1-hhhh)

The format definition uses 'h' as a hexadecimal unit. Uppercase characters and all other symbols including the space character between parentheses appear as shown.

[Click on this highlighted text to access the reference file.](#)

NOTE – The content is also available as a separate viewable file in the same file directory as this document. The file is named: "CJKC0SR.txt".

28 Character names and annotations

28.1 General

Guidelines to be used for constructing names of characters are given in annex L for information. In some cases, a name of a character is followed by additional explanatory statements not part of the name. These

statements are in parentheses and not in capital letters except for the initials of the word, where required.

28.2 Character names for CJK Ideographs

For CJK Ideographs the names are algorithmically constructed by appending their coded representation in hexadecimal notation to “CJK UNIFIED IDEOGRAPH-” for CJK Unified Ideographs and “CJK COMPATIBILITY IDEOGRAPH-” for CJK Compatibility Ideographs.

For CJK Ideographs within the BMP, the coded representation is their two-octet value. For example, the first CJK Ideograph character in the BMP has the name “CJK UNIFIED IDEOGRAPH-3400”.

For CJK Ideographs within the SIP, the coded representation is their five-digit value. For example, the first CJK Ideograph character in the SIP has the name “CJK UNIFIED IDEOGRAPH-20000”.

28.3 Character names and annotations for Hangul syllables

Names for the Hangul syllable characters in code positions 0000 AC00 - 0000 D7A3 are derived from their code position numbers by the numerical procedure described below. Lists of names for these characters are not provided opposite the code tables.

1. Obtain the code position number of the Hangul syllable character. It is of the form 0000 $h_1h_2h_3h_4$ where h_1 , h_2 , h_3 , and h_4 are hexadecimal digits; h_1h_2 is the Row number within the BMP and h_3h_4 is the cell number within the row. The number $h_1h_2h_3h_4$ lies within the range AC00 to D7A3.

2. Derive the decimal numbers d_1 , d_2 , d_3 , d_4 that are numerically equal to the hexadecimal digits h_1 , h_2 , h_3 , h_4 respectively.

3. Calculate the character index C from the formula:

$$C = 4096 \times (d_1 - 10) + 256 \times (d_2 - 12) + 16 \times d_3 + d_4$$

NOTE – If $C < 0$ or > 11171 then the character is not in the HANGUL SYLLABLES block.

4. Calculate the syllable component indices I , P , F from the following formulae:

$$I = C / 588 \quad (\text{Note: } 0 \leq I \leq 18)$$

$$P = (C \% 588) / 28 \quad (\text{Note: } 0 \leq P \leq 20)$$

$$F = C \% 28 \quad (\text{Note: } 0 \leq F \leq 27)$$

where “/” indicates integer division (i.e. x / y is the integer quotient of the division), and “%” indicates the modulo operation (i.e. $x \% y$ is the remainder after the integer division x / y).

5. Obtain the Latin character strings that correspond to the three indices I , P , F from columns 2, 3, and 4 respectively of table 1 below (for $I = 11$ and for $F = 0$ the corresponding strings are null). Concatenate these three strings in left-to-right order to make a single string, the syllable-name.

6. The character name for the character at position 0000 $h_1h_2h_3h_4$ is then:

HANGUL SYLLABLE $s-n$

where “ $s-n$ ” indicates the syllable-name string derived in step 5.

Example.

For the character in code position D4DE:

$$d_1 = 13, d_2 = 4, d_3 = 13, d_4 = 14.$$

$$C = 10462$$

$$I = 17, P = 16, F = 18.$$

The corresponding Latin character strings are:

P, WI, BS.

The syllable-name is PWIBS, and the character name is:

HANGUL SYLLABLE PWIBS

For convenience a list of the character names of the Hangul syllables is provided in annex R.

For each Hangul syllable character a short annotation is defined. This annotation consists of an alternative transliteration of the Hangul syllable into Latin characters.

Annotations for the Hangul syllable characters in code positions 0000 AC00 - 0000 D7A3 are also derived from their code position numbers by a similar numerical procedure described below.

7. Carry out steps 1 to 4 as described above.

8. Obtain the Latin character strings that correspond to the three indices I , P , F from columns 5, 6, and 7 respectively of Table 1 below (for $I = 11$ and for $F = 0$ the corresponding strings are null). Concatenate these three strings in left-to-right order to make a single string, and enclose it within parentheses to form the annotation.

Example.

For the character in code position D4DE:

$$d_1 = 13, d_2 = 4, d_3 = 13, d_4 = 14.$$

$$C = 10462$$

$$I = 17, P = 16, F = 18.$$

The corresponding Latin character strings are:

ph, wi, ps,

and the annotation is (phwips).

Table 1: Elements of Hangul syllable names and annotations

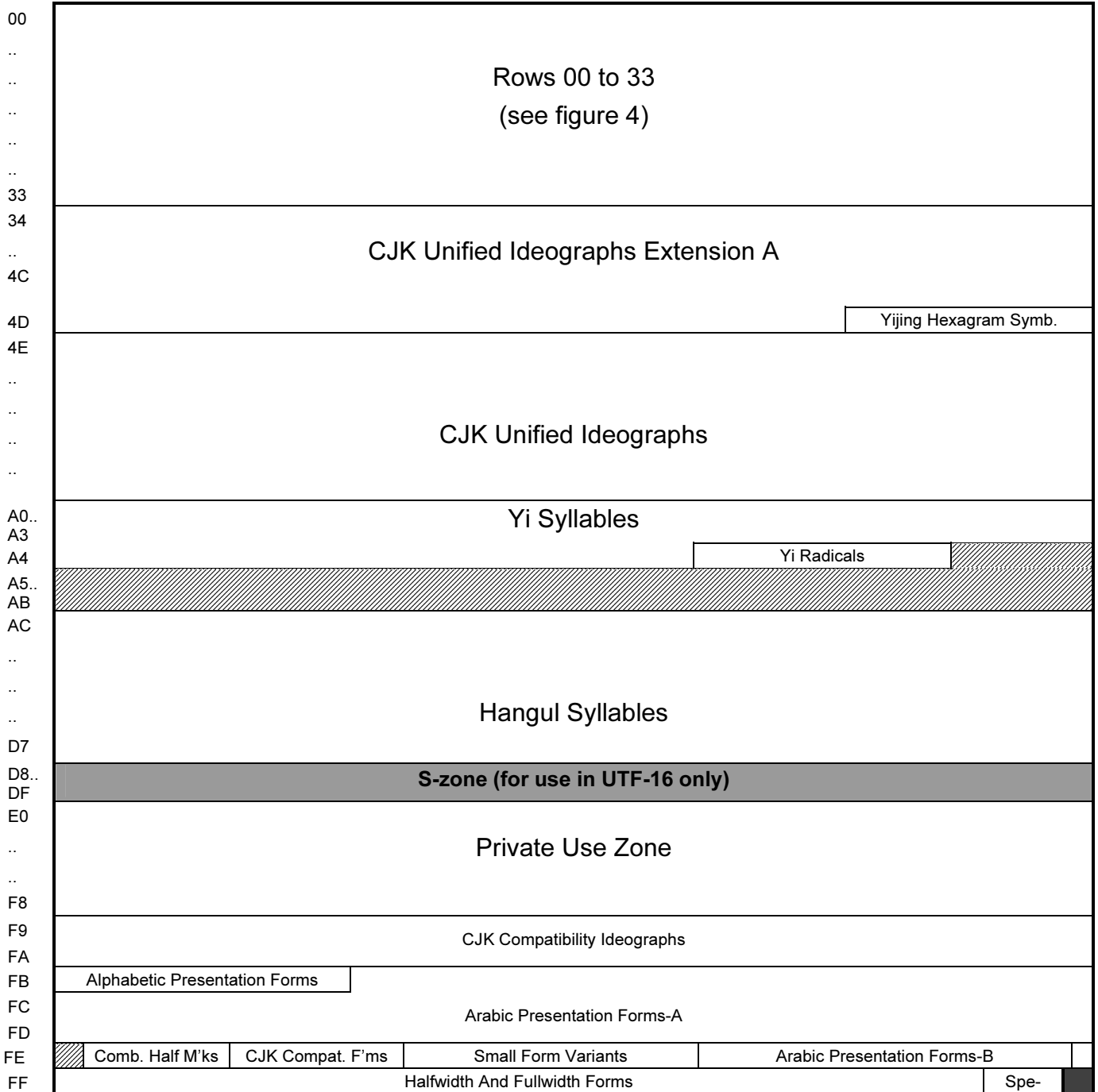
Index number	Syllable name elements			Annotation elements		
	<i>I</i> string	<i>P</i> string	<i>F</i> string	<i>I</i> string	<i>P</i> string	<i>F</i> string
0	G	A		k	a	
1	GG	AE	G	kk	ae	k
2	N	YA	GG	n	ya	kk
3	D	YAE	GS	t	yae	ks
4	DD	EO	N	tt	eo	n
5	R	E	NJ	r	e	nc
6	M	YEO	NH	m	yeo	nh
7	B	YE	D	p	ye	t
8	BB	O	L	pp	o	l
9	S	WA	LG	s	wa	lk
10	SS	WAE	LM	ss	wae	lm
11		OE	LB		oe	lp
12	J	YO	LS	c	yo	ls
13	JJ	U	LT	cc	u	lth
14	C	WEO	LP	ch	weo	lph
15	K	WE	LH	kh	we	lh
16	T	WI	M	th	wi	m
17	P	YU	B	ph	yu	p
18	H	EU	BS	h	eu	ps
19		YI	S		yi	s
20		I	SS		i	ss
21			NG			ng
22			J			c
23			C			ch
24			K			kh
25			T			th
26			P			ph
27			H			h

29 Structure of the Basic Multilingual Plane

An overview of the Basic Multilingual Plane is shown in figure 3 and a more detailed overview of Rows 00 to 33 is shown in figure 4.

The Basic Multilingual Plane includes characters in general use in alphabetic, syllabic, and ideographic scripts together with various symbols and digits.

Row-octet




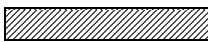
= not graphic characters
 = reserved for future standardization

NOTE - Vertical boundaries within rows are indicated in approximate positions only.

Figure 3 - Overview of the Basic Multilingual Plane

Row-octet

00	Basic Latin				Latin-1 Supplement			
01	Latin Extended-A				Latin Extended-B			
02	Latin Extended-B		IPA (Intl. Phonetic Alph.) Extensions		Spacing Modifier Letters			
03	Combining Diacritical Marks			Greek and Coptic				
04	Cyrillic							
05	Cyrillic Supplement		Armenian			Hebrew		
06	Arabic							
07	Syriac				Thaana			
08								
09	Devanagari				Bengali			
0A	Gurmukhi				Gujarati			
0B	Oriya				Tamil			
0C	Telugu				Kannada			
0D	Malayalam				Sinhala			
0E	Thai				Lao			
0F	Tibetan							
10	Myanmar				Georgian			
11	Hangul Jamo							
12	Ethiopic							
13					Cherokee			
14	Unified Canadian Aboriginal Syllabics							
16					Ogham		Runic	
17	Tagalog	Hanunoo	Buhid	Tagbanwa	Khmer			
18	Mongolian							
19	Limbu		Tai Le		Khmer Symb.			
1A..								
1C								
1D	Phonetic Extension							
1E	Latin Extended Additional							
1F	Greek Extended							
20	General Punctuation			Super-/Subscripts		Currency Symbols		Comb. Mks. Symb.
21	Letterlike Symbols		Number Forms		Arrows			
22	Mathematical Operators							
23	Miscellaneous Technical							
24	Control Pictures		O.C.R.	Enclosed Alphanumerics				
25	Box Drawing			Block Elements		Geometric Shapes		
26	Miscellaneous Symbols							
27	Dingbats					Misc. Math.Symb.-A	S. Arrows-A	
28	Braille Patterns							
29	Supplemental Arrows-B				Miscellaneous Mathematical Symbols-B			
2A	Supplemental Mathematical Operators							
2B	Miscellaneous Symbols and Arrows							
2C..								
2D								
2E					CJK Radicals Supplement			
2F	Kangxi Radicals						Ideog. Descr.	
30	CJK Symbols And Punctuation		Hiragana			Katakana		
31	Bopomofo	Hangul Compatibility Jamo		Kanbun	Bopomofo Ext.		K. P.E.	
32	Enclosed CJK Letters And Months							
33	CJK Compatibility							

 = not graphic characters
  = reserved for future standardization

NOTE - Vertical boundaries within rows are indicated in approximate positions only.

Figure 4 - Overview of Rows 00 to 33 of the Basic Multilingual Plane

30 Structure of the Supplementary Multilingual Plane for Scripts and symbols

Because another supplementary plane is reserved for additional CJK Ideographs, the SMP is not used to encode any CJK Ideographs. The SMP is scheduled to contain coded graphic characters used in other scripts of the world that are not encoded or not already scheduled for encoding in the BMP. Most, but not all, of the scripts encoded or scheduled for encoding in the SMP are not in use as living scripts by modern user communities.


NOTE - The following subdivision of the SMP has been proposed:

- Alphabetic scripts,
- Hieroglyphic, ideographic and syllabaries,
- Non CJK ideographic scripts,
- Newly invented scripts,
- Symbol sets.

An overview of the Supplementary Multilingual Plane for scripts and symbols is shown in figure 5.

Row-octet

00	Linear B Syllabary		Linear B Ideograms	
01	Aegean Numbers			
...				
03	Old Italic	Gothic	Ugaritic	
04	Deseret		Shavian	Osmanya
...				
08	Cypriot Syllabary			
D0	Byzantine Musical Symbols			
D1	Western Musical Symbols			
...				
D3	Tai Xuan Jing Symbols			
D4	Mathematical Alphanumeric Symbols			
...				
D7				
...				
FF				

 = reserved for future standardization

NOTE 1 - Vertical boundaries within rows are indicated in approximate positions only.

NOTE 2 - The Old Italic block represents a unified script that covers the Etruscan, Oscan, Umbrian, Faliscan, North Picene, and South Picene alphabets. Some of these alphabets can be written with characters oriented in either left-to-right or right-to-left direction. The glyphs in the code table are shown with left to right orientation.

Figure 5 – Overview of the Supplementary Multilingual Plane for scripts and symbols

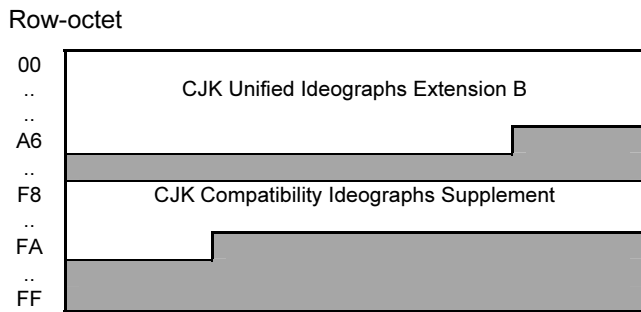
31 Structure of the Supplementary Ideographic Plane

The Plane 02 of Group 00 is the Supplementary Ideographic Plane (SIP).

The SIP is used for CJK unified ideographs (unified East Asian ideographs) that are not encoded in the BMP. The procedure for the unification and arrangement of the SIP CJK unified Ideographs is described in clause 27.1.

The SIP is also used for compatibility CJK ideographs. These ideographs are compatibility characters as specified in clause 4.13.

The following figure 6 shows an overview of the Supplementary Ideographic Plane.



[Grey box] = reserved for future standardization
 NOTE - Vertical boundaries within rows are indicated in approximate positions only.

Figure 6 – Overview of the Supplementary Ideographic Plane

32 Structure of the Supplementary Special-purpose Plane

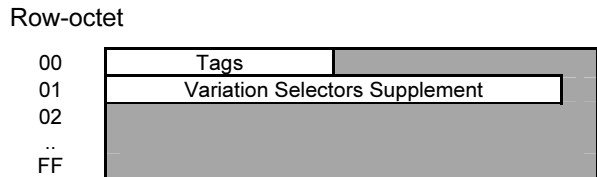
The Plane 0E of Group 0 is the Supplementary Special-purpose Plane (SSP).

The SSP is used for special purpose use graphic characters. Code positions from E0000 to E0FFF are reserved for Alternate Format Characters (see 20).

NOTE - Some of these characters do not have a visual representation and do not have printable graphic symbols. The Tag Characters are example of such characters.

An overview of the Supplementary Special-purpose Plane is shown in figure 7.

NOTE - Unassigned code points in this range should be ignored in normal processing and display.



[Grey box] = reserved for future standardization
 NOTE - Vertical boundaries within rows are indicated in approximate positions only.

Figure 7 – Overview of the Supplementary Special-purpose Plane

33 Code tables and lists of character names

Detailed code tables and lists of character names for the BMP, SMP, SIP and SSP are shown on the following pages.

[Charts]

*Tables of character graphic symbols for all Planes
will appear on this and following pages in the Final Text.
(total xxx pages numbered aaa to bbb)*

[Charts]

*Tables of character graphic symbols for all Planes
will appear on this and following pages in the Final Text.
(total xxx pages numbered aaa to bbb)*

Annex A (normative)

Collections of graphic characters for subsets

A.1 Collections of coded graphic characters

The collections listed below are ordered by collection number. An * in the “positions” column indicates that the collection is a fixed collection.

<u>Collection number and name</u>	<u>Positions</u>		
1 BASIC LATIN	0020 - 007E *	22 TELUGU	0C00 - 0C7F 200C, 200D
2 LATIN-1 SUPPLEMENT	00A0 - 00FF *	23 KANNADA	0C80 - 0CFF 200C, 200D
3 LATIN EXTENDED-A	0100 - 017F *	24 MALAYALAM	0D00 - 0D7F 200C, 200D
4 LATIN EXTENDED-B	0180 - 024F	25 THAI	0E00 - 0E7F
5 IPA EXTENSIONS	0250 - 02AF	26 LAO	0E80 - 0EFF
6 SPACING MODIFIER LETTERS	02B0 - 02FF *	27 BASIC GEORGIAN	10D0 - 10FF
7 COMBINING DIACRITICAL MARKS	0300 - 036F	28 GEORGIAN EXTENDED	10A0 - 10CF
8 BASIC GREEK	0370 - 03CF	29 HANGUL JAMO	1100 - 11FF
9 GREEK SYMBOLS AND COPTIC	03D0 - 03FF	30 LATIN EXTENDED ADDITIONAL	1E00 - 1EFF
10 CYRILLIC	0400 - 04FF	31 GREEK EXTENDED	1F00 - 1FFF
11 ARMENIAN	0530 - 058F	32 GENERAL PUNCTUATION	2000 - 206F
12 BASIC HEBREW	05D0 - 05EA *	33 SUPERSCRIPTS AND SUBSCRIPTS	2070 - 209F
13 HEBREW EXTENDED	0590 - 05CF 05EB - 05FF	34 CURRENCY SYMBOLS	20A0 - 20CF
14 BASIC ARABIC	0600 - 065F	35 COMBINING DIACRITICAL MARKS FOR SYMBOLS	20D0 - 20FF
15 ARABIC EXTENDED	0660 - 06FF *	36 LETTERLIKE SYMBOLS	2100 - 214F
16 DEVANAGARI	0900 - 097F 200C, 200D	37 NUMBER FORMS	2150 - 218F
17 BENGALI	0980 - 09FF 200C, 200D	38 ARROWS	2190 - 21FF *
18 GURMUKHI	0A00 - 0A7F 200C, 200D	39 MATHEMATICAL OPERATORS	2200 - 22FF *
19 GUJARATI	0A80 - 0AFF 200C, 200D	40 MISCELLANEOUS TECHNICAL	2300 - 23FF
20 ORIYA	0B00 - 0B7F 200C, 200D	41 CONTROL PICTURES	2400 - 243F
21 TAMIL	0B80 - 0BFF 200C, 200D	42 OPTICAL CHARACTER RECOGNITION	2440 - 245F
		43 ENCLOSED ALPHANUMERICS	2460 - 24FF *
		44 BOX DRAWING	2500 - 257F *
		45 BLOCK ELEMENTS	2580 - 259F *
		46 GEOMETRIC SHAPES	25A0 - 25FF *
		47 MISCELLANEOUS SYMBOLS	2600 - 26FF

48	DINGBATS	2700 - 27BF	81	CJK UNIFIED IDEOGRAPHS EXTENSION A	3400 - 4DBF FA1F, FA23
49	CJK SYMBOLS AND PUNCTUATION	3000 - 303F *	82	OGHAM	1680 - 169F
50	HIRAGANA	3040 - 309F	83	RUNIC	16A0 - 16FF
51	KATAKANA	30A0 - 30FF *	84	SINHALA	0D80 - 0DFF
52	BOPOMOFO	3100 - 312F 31A0 - 31BF	85	SYRIAC	0700 - 074F
53	HANGUL COMPATIBILITY JAMO	3130 - 318F	86	THAANA	0780 - 07BF
54	CJK MISCELLANEOUS	3190 - 319F	87	BASIC MYANMAR	1000 - 104F 200C, 200D
55	ENCLOSED CJK LETTERS AND MONTHS	3200 - 32FF	88	KHMER	1780 - 17FF 200C, 200D
56	CJK COMPATIBILITY	3300 - 33FF *	89	MONGOLIAN	1800 - 18AF
57, 58, 59	(These collection numbers shall not be used, see Note 2.)		90	EXTENDED MYANMAR	1050 - 109F
60	CJK UNIFIED IDEOGRAPHS	4E00 - 9FFF	91	TIBETAN	0F00 - 0FFF
61	PRIVATE USE AREA	E000 - F8FF	92	CYRILLIC SUPPLEMENT	0500 - 052F
62	CJK COMPATIBILITY IDEOGRAPHS	F900 - FAFF	93	TAGALOG	1700 - 171F
63	(Collection specified as union of other collections)		94	HANUNOO	1720 - 173F
64	ARABIC PRESENTATION FORMS-A	FB50 - FDCF FDF0 - FDFE	95	BUHID	1740 - 175F
65	COMBINING HALF MARKS	FE20 - FE2F	96	TAGBANWA	1760 - 177F
66	CJK COMPATIBILITY FORMS	FE30 - FE4F *	97	MISCELLANEOUS MATHEMATICAL SYMBOLS-A	27C0 - 27EF
67	SMALL FORM VARIANTS	FE50 - FE6F	98	SUPPLEMENTAL ARROWS-A	27F0 - 27FF *
68	ARABIC PRESENTATION FORMS-B	FE70 - FEFE	99	SUPPLEMENTAL ARROWS-B	2900 - 297F *
69	HALFWIDTH AND FULLWIDTH FORMS	FF00 - FFEF	100	MISCELLANEOUS MATHEMATICAL SYMBOLS-B	2980 - 29FF *
70	SPECIALS	FFF0 - FFFD	101	SUPPLEMENTAL MATHEMATICAL OPERATORS	2A00 - 2AFF *
71	HANGUL SYLLABLES	AC00 - D7A3 *	102	KATAKANA PHONETIC EXTENSIONS	31F0 - 31FF *
72	BASIC TIBETAN	0F00 - 0FBF	103	VARIATION SELECTORS	FE00 - FE0F *
73	ETHIOPIC	1200 - 137F	104	LTR ALPHABETIC PRESENTATION FORMS	FB00 - FB1C
74	UNIFIED CANADIAN ABORIGINAL SYLLABICS	1400 - 167F	105	RTL ALPHABETIC PRESENTATION FORMS	FB1D - FB4F
75	CHEROKEE	13A0 - 13FF	106	LIMBU	1900 - 194F
76	YI SYLLABLES	A000 - A48F	107	TAI LE	1950 - 197F
77	YI RADICALS	A490 - A4CF	108	KHMER SYMBOLS	19E0 - 19FF *
78	KANGXI RADICALS	2F00 - 2FDF	109	PHONETIC EXTENSIONS	1D00 - 1D7F
79	CJK RADICALS SUPPLEMENT	2E80 - 2EFF	110	MISCELLANEOUS SYMBOLS AND ARROWS	2B00-2B7F
80	BRAILLE PATTERNS	2800 - 28FF	111	YIJING HEXAGRAM SYMBOLS	4DC0-4DDD *
			1001	OLD ITALIC	10300-1032F

1002	GOTHIC	10330-1034F
1003	DESERET	10400-1044F*
1004	BYZANTINE MUSICAL SYMBOLS	1D000-1D0FF
1005	MUSICAL SYMBOLS	1D100-1D1FF
1006	MATHEMATICAL ALPHANUMERIC SYMBOLS	1D400-1D7FF
1007	LINEAR B SYLLABARY	10000-1007F
1008	LINEAR B IDEOGRAMS	10080-100FF
1009	AEGEAN NUMBERS	10100-1013F
1010	UGARITIC	10380-1039F
1011	SHAVIAN	10450-1047F*
1012	OSMANYA	10480-104AF
1013	CYPRIOT SYLLABARY	10800-1083F
1014	TAI XUAN JING SYMBOLS	1D300-1D35F
2001	CJK UNIFIED IDEOGRAPHS EXTENSION B	20000-2A6DF
2002	CJK COMPATIBILITY IDEOGRAPHS SUPPLEMENT	2F800-2FA1F
3001	TAGS	E0000-E007F
3003	VARIATION SELECTORS SUPPLEMENT	E0100-E01EF*

The following collections specify characters used for alternate formats and script-specific formats. See annex F for more information.

200	ZERO-WIDTH BOUNDARY INDICATORS	200B - 200D FEFF
201	FORMAT SEPARATORS	2028 - 2029
202	BI-DIRECTIONAL FORMAT MARKS	200E - 200F
203	BI-DIRECTIONAL FORMAT EMBEDDINGS	202A - 202E
204	HANGUL FILL CHARACTERS	3164, FFA0
205	CHARACTER SHAPING SELECTORS	206A - 206D
206	NUMERIC SHAPE SELECTORS	206E - 206F
207	IDEOGRAPHIC DESCRIPTION CHARACTERS	2FF0 - 2FFF
3002	ALTERNATE FORMAT CHARACTERS	E0000-E0FFF

The following specify other collections.

270	COMBINING CHARACTERS	characters specified in annex B.1
271	COMBINING CHARACTERS B-2	characters specified in annex B.2
281	MES-1	see A.4.1 *
282	MES-2	see A.4.2 *
283	MODERN EUROPEAN SCRIPTS	see A.4.3 *
299	(This collection number shall not be used, see A.3.2.)	
300	BMP	0000 - D7FF E000 - FFFD
301	BMP-AMD.7	see A.3.1 *
302	BMP SECOND EDITION	see A.3.3 *
1000	SMP	10000-1FFFFD
1900	SMP COMBINING CHARACTERS	characters specified in annex B.1
2000	SIP	20000-2FFFFD
3000	SSP	E0000-EFFFFD

The following specify collections which are the union of particular collections defined above.

63	ALPHABETIC PRESENTATION FORMS	Collections 104-105
250	GENERAL FORMAT CHARACTERS	Collections 200 - 203
251	SCRIPT-SPECIFIC FORMAT CHARACTERS	Collections 204 - 206
4000	UCS PART-2	Collections 1000, 2000, 3000

The following collections contain characters both inside and outside the Basic Multilingual Plane.

303	UNICODE 3.1	see A5.1 *
304	UNICODE 3.2	see A5.2 *
305	UNICODE 4.0	see A5.3 *
380	CJK UNIFIED IDEOGRAPHS-2001 *	3400-4DB5 4E00-9FA5 FA0E-FA0F FA11 FA13-FA14 FA1F FA21 FA23-FA24 FA27-FA29 20000-2A6D6

381 CJK COMPATIBILITY IDEOGRAPHS-2001 *
 F900-FA0D
 FA10
 FA12
 FA15-FA1E
 FA20
 FA22
 FA25-FA26
 FA2A-FA6A
 2F800-2FA1D

10646 UNICODE
 0000-FDCF
 FDF0-FFFFD
 10000-1FFFFD
 20000-2FFFFD
 30000-3FFFFD
 40000-4FFFFD
 50000-5FFFFD
 60000-6FFFFD
 70000-7FFFFD
 80000-8FFFFD
 90000-9FFFFD
 A0000-AFFFFD
 B0000-BFFFFD
 C0000-CFFFFD
 D0000-DFFFFD
 E0000-EFFFFD
 F0000-FFFFFD
 100000-10FFFFD

NOTE – The UNICODE collection incorporates all characters currently encoded in the standard.

The following collections are outside the Basic Multilingual Plane.

400 (This collection number shall not be used, see Note 2.)

401 PRIVATE USE PLANES-0F-10 G=00,
 P=0F-10

500 (This collection number shall not be used, see Note 2.)

NOTE 1 - Use of implementation levels 1 and 2 restricts the repertoire of some character collections (see 24.4). Collections which include combining characters are 7, 10, 13 to 26, 35, 49, 50, 63, 65, 72, 84, 85, 86, 87, 88, 89, 90, 91, 93, 94, 95, 96, 104 and 1005.

NOTE 2 - Collections numbered 57, 58, and 59 were specified in the First Edition of ISO/IEC 10646-1 but have now been deleted. Collections numbered 400 and 500 were specified in the First and Second Editions of ISO/IEC 10646-1 but have now been deleted.

NOTE 3 - The principal terms (keywords) used in the collection names shown above are listed below in alphabetical order. The entry for a term shows the collection number of every collection whose name includes the term. These terms do not provide a complete cross-reference to all the collections where characters sharing a particular attribute, such as script name, may be found. Although most of the terms identify an attribute of the characters within the collection, some characters that possess that attribute may be

present in other collections whose numbers do not appear in the entry for that term.

Aegean numbers	1009
Alphabetic	63
Alphanumeric	43
Arabic	14 15 64 68
Armenian	11
Arrows	38 98 99 110
Bengali	17
Bidirectional	202 203
Block elements	45
BMP	300 301 302 (299)
Box drawing	44
Bopomofo	52
Braille patterns	80
Buhid	95
Byzantine musical symbols	1004
Canadian Aboriginal	74
Cherokee	75
CJK	49 54 55 56 60 62 66 78 81 2001 2002
Combining	7 35 65 270 271
Compatibility	53 56 62 66
Control pictures	41
Coptic	9
Currency	34
Cypriot syllabary	1013
Cyrillic	10 92
Deseret	1003
Devanagari	16
Diacritical marks	7 35
Dingbats	48
Enclosed	43 55
Ethiopic	73
Format	201 202 203 250 251
Fullwidth	69
Geometric shapes	46
Georgian	27 28
Gothic	1002
Greek	8 9 31
Gujarati	19
Gurmukhi	18
Half (marks, width)	65 69
Hangul	29 53 71 204
Hanunoo	94
Hebrew	12 13
Hiragana	50
Ideographs	60 62 81 207 380 381
IPA extensions	5
Jamo	29 53
Kangxi	78
Kannada	23
Katakana	51 102
Khmer	88 108
Lao	26
Latin	1 2 3 4 30
Letter	36 55
Limbu	106
Linear B syllabary	1007
Linear B ideograms	1008
Malayalam	24
Mathematical alphanumeric symbols	1006
Mathematical operators	39 101

Mathematical symbols	97 100	CYRILLIC	0400 - 04FF
MES	281 282	CYRILLIC SUPPLEMENT	0500 - 052F
Mongolian	89	ARMENIAN	0530 - 058F
Months	55	HEBREW	0590 - 05FF
Musical symbols	1005	ARABIC	0600 - 06FF
Myanmar	87 90	SYRIAC	0700 - 074F
Number	37	THAANA	0780 - 07BF
Ogham	82	DEVANAGARI	0900 - 097F
Old Italic	1001	BENGALI	0980 - 09FF
Optical character recognition	42	GURMUKHI	0A00 - 0A7F
Oriya	20	GUJARATI	0A80 - 0AFF
Osmanya	1012	ORIYA	0B00 - 0B7F
Phonetic extensions	109	TAMIL	0B80 - 0BFF
Presentation forms	63 64 68 104 105	TELUGU	0C00 - 0C7F
Private use	61 401	KANNADA	0C80 - 0CFF
Punctuation	32 49	MALAYALAM	0D00 - 0D7F
Radicals	77 78 79	SINHALA	0D80 - 0DFF
Runic	83	THAI	0E00 - 0E7F
Shape, shaping	205 206	LAO	0E80 - 0EFF
Shavian	1011	TIBETAN	0F00 - 0FFF
Sinhala	84	MYANMAR	1000 - 109F
Small form	67	GEORGIAN	10A0 - 10FF
Spacing modifier	6	HANGUL JAMO	1100 - 11FF
Specials	70	ETHIOPIC	1200 - 137F
Subscripts, superscripts	33	CHEROKEE	13A0 - 13FF
Syllables, syllabics	71 74 76	UNIFIED CANADIAN ABORIGINAL SYLLABICS	1400 - 167F
Symbols	9 34 35 36 47 49 97 100	OGHAM	1680 - 169F
Syriac	85	RUNIC	16A0 - 16FF
Tagalog	93	TAGALOG	1700 - 171F
Tagbanwa	96	HANUNOO	1720 - 173F
Tags	3001	BUHID	1740 - 175F
Tai Xuan Jing symbols	1014	TAGBANWA	1760 - 177F
Tail Le	107	KHMER	1780 - 17FF
Tamil	21	MONGOLIAN	1800 - 18AF
Technical	40	LIMBU	1900 - 194F
Telugu	22	TAI LE	1950 - 197F
Thaana	86	KHMER SYMBOLS	19E0 - 19FF
Thai	25	PHONETIC EXTENSIONS	1D00 - 1D7F
Tibetan	72 91	LATIN EXTENDED ADDITIONAL	1E00 - 1EFF
Ugaritic	1010	GREEK EXTENDED	1F00 - 1FFF
Unicode	303 304 10646	GENERAL PUNCTUATION	2000 - 206F
Variation selectors	103 3003	SUPERSCRIPTS AND SUBSCRIPTS	2070 - 209F
Yi	76 77	CURRENCY SYMBOLS	20A0 - 20CF
Yijing hexagram symbols	111	COMBINING DIACRITICAL MARKS FOR SYMBOLS	20D0 - 20FF
Zero-width	200	LETTERLIKE SYMBOLS	2100 - 214F
		NUMBER FORMS	2150 - 218F
		ARROWS	2190 - 21FF
		MATHEMATICAL OPERATORS	2200 - 22FF
		MISCELLANEOUS TECHNICAL	2300 - 23FF
		CONTROL PICTURES	2400 - 243F
		OPTICAL CHARACTER RECOGNITION	2440 - 245F
		ENCLOSED ALPHANUMERIC	2460 - 24FF
		BOX DRAWING	2500 - 257F
		BLOCK ELEMENTS	2580 - 259F
		GEOMETRIC SHAPES	25A0 - 25FF
		MISCELLANEOUS SYMBOLS	2600 - 26FF
		DINGBATS	2700 - 27BF
		MISCELLANEOUS MATHEMATICAL SYMBOLS-A	27C0 - 27EF
		SUPPLEMENTAL ARROWS-A	27F0 - 27FF
		BRAILLE PATTERNS	2800 - 28FF
		SUPPLEMENTAL ARROWS-B	2900 - 297F

A.2 Blocks lists

A.2.1 Blocks in the BMP

The following blocks are specified in the Basic Multilingual Plane. They are ordered by code position.

<u>Block name</u>	<u>from</u>	<u>to</u>
BASIC LATIN	0020	007E
LATIN-1 SUPPLEMENT	00A0	00FF
LATIN EXTENDED-A	0100	017F
LATIN EXTENDED-B	0180	024F
IPA (INTERNATIONAL PHONETIC ALPHABET) EXTENSIONS	0250	02AF
SPACING MODIFIER LETTERS	02B0	02FF
COMBINING DIACRITICAL MARKS	0300	036F
GREEK AND COPTIC	0370	03FF

MISCELLANEOUS MATHEMATICAL SYMBOLS-B	2980 - 29FF
SUPPLEMENTAL MATHEMATICAL OPERATORS	2A00 - 2AFF
MISCELLANEOUS SYMBOLS AND ARROWS	2B00 - 2B7F
CJK RADICALS SUPPLEMENT	2E80 - 2EFF
KANGXI RADICALS	2F00 - 2FDF
IDEOGRAPHIC DESCRIPTION CHARACTERS	2FF0 - 2FFF
CJK SYMBOLS AND PUNCTUATION	3000 - 303F
HIRAGANA	3040 - 309F
KATAKANA	30A0 - 30FF
BOPOMOFO	3100 - 312F
HANGUL COMPATIBILITY JAMO	3130 - 318F
KANBUN (CJK miscellaneous)	3190 - 319F
BOPOMOFO EXTENDED	31A0 - 31BF
KATAKANA PHONETIC EXTENSIONS	31F0 - 31FF
ENCLOSED CJK LETTERS AND MONTHS	3200 - 32FF
CJK COMPATIBILITY	3300 - 33FF
CJK UNIFIED IDEOGRAPHS EXTENSION A	3400 - 4DBF
YIJING HEXAGRAM SYMBOLS	4DC0 - 4DFF
CJK UNIFIED IDEOGRAPHS	4E00 - 9FFF
YI SYLLABLES	A000 - A48F
YI RADICALS	A490 - A4CF
HANGUL SYLLABLES	AC00 - D7A3
PRIVATE USE AREA	E000 - F8FF
CJK COMPATIBILITY IDEOGRAPHS	F900 - FAFF
ALPHABETIC PRESENTATION FORMS	FB00 - FB4F
ARABIC PRESENTATION FORMS-A	FB50 - FDFF
VARIATION SELECTORS	FE00 - FE0F
COMBINING HALF MARKS	FE20 - FE2F
CJK COMPATIBILITY FORMS	FE30 - FE4F
SMALL FORM VARIANTS	FE50 - FE6F
ARABIC PRESENTATION FORMS-B	FE70 - FEFE
HALFWIDTH AND FULLWIDTH FORMS	FF00 - FFEF
SPECIALS	FFF0 - FFFF

A.2.2 Blocks in the SMP

The following blocks are specified in the Supplementary Multilingual Plane for scripts and symbols. They are ordered by code position.

<u>Block name</u>	<u>from</u> <u>to</u>
LINEAR B SYLLABARY	10000-1007F
LINEAR B IDEOGRAMS	10080-100FF
AEGEAN NUMBERS	10100-1013F
OLD ITALIC	10300-1032F
GOTHIC	10330-1034F
UGARITIC	10380-1039F
DESERET	10400-1044F
SHAVIAN	10450-1047F
OSMANYA	10480-104AF
CYPRIOT SYLLABARY	10800-1083F
BYZANTINE MUSICAL SYMBOLS	1D000-1D0FF
MUSICAL SYMBOLS	1D100-1D1FF
TAI XUAN JING SYMBOLS	1D300-1D35F
MATHEMATICAL ALPHANUMERIC SYMBOLS	1D400-1D7FF

A.2.3 Blocks in the SIP

The following blocks are specified in the Supplementary Ideographic Plane. They are ordered by code position.

<u>Block name</u>	<u>from</u> <u>to</u>
CJK UNIFIED IDEOGRAPHS EXTENSION B	20000-2A6DF
CJK COMPATIBILITY IDEOGRAPHS SUPPLEMENT	2F800-2FA1F

A.2.4 Blocks in the SSP

The following blocks are specified in the Supplementary Special-purpose Plane. They are ordered by code position.

<u>Block name</u>	<u>from</u> <u>to</u>
TAGS	E0000-E007F
VARIATION SELECTORS SUPPLEMENT	E0100-E01EF

A.3 Fixed collections of the whole BMP

A.3.1 301 BMP-AMD.7

The collection 301 BMP-AMD.7 is specified below as a fixed collection (4.19). It comprises only those coded characters that were in the BMP after amendments up to, but not after, AMD.7 were applied to the First Edition of ISO/IEC 10646-1. Accordingly the repertoire of this collection is not subject to change if new characters are added to the BMP by any subsequent amendments.

NOTE - The repertoire of the collection 300 BMP is subject to change if new characters are added to the BMP by an amendment to this International Standard.

301 BMP-AMD.7 is specified by the following ranges of code positions as indicated for each row or contiguous series of rows.

<u>Rows</u>	<u>Positions (cells)</u>
00	20-7E A0-FF
01	00-F5 FA-FF
02	00-17 50-A8 B0-DE E0-E9
03	00-45 60-61 74-75 7A 7E 84-8A 8C 8E-A1 A3-CE D0-D6 DA DC DE E0 E2-F3
04	01-0C 0E-4F 51-5C 5E-86 90-C4 C7-C8 CB-CC D0-EB EE-F5 F8-F9
05	31-56 59-5F 61-87 89 91-A1 A3-B9 BB-C4 D0-EA F0-F4
06	0C 1B 1F 21-3A 40-52 60-6D 70-B7 BA-BE C0-CE D0-ED F0-F9
09	01-03 05-39 3C-4D 50-54 58-70 81-83 85-8C 8F-90 93-A8 AA-B0 B2 B6-B9 BC BE-C4 C7-C8 CB-CD D7 DC-DD DF-E3 E6-FA
0A	02 05-0A 0F-10 13-28 2A-30 32-33 35-36 38-39 3C 3E-42 47-48 4B-4D 59-5C 5E 66-74 81-83 85-8B 8D 8F-91 93-A8 AA-B0 B2-B3 B5-B9 BC-C5 C7-C9 CB-CD D0 E0 E6-EF

0B	01-03 05-0C 0F-10 13-28 2A-30 32-33 36-39 3C-43 47-48 4B-4D 56-57 5C-5D 5F-61 66- 70 82-83 85-8A 8E-90 92-95 99-9A 9C 9E- 9F A3-A4 A8-AA AE-B5 B7-B9 BE-C2 C6- C8 CA-CD D7 E7-F2
0C	01-03 05-0C 0E-10 12-28 2A-33 35-39 3E- 44 46-48 4A-4D 55-56 60-61 66-6F 82-83 85-8C 8E-90 92-A8 AA-B3 B5-B9 BE-C4 C6-C8 CA-CD D5-D6 DE E0-E1 E6-EF
0D	02-03 05-0C 0E-10 12-28 2A-39 3E-43 46- 48 4A-4D 57 60-61 66-6F
0E	01-3A 3F-5B 81-82 84 87-88 8A 8D 94-97 99-9F A1-A3 A5 A7 AA-AB AD-B9 BB-BD C0-C4 C6 C8-CD D0-D9 DC-DD
0F	00-47 49-69 71-8B 90-95 97 99-AD B1-B7 B9
10	A0-C5 D0-F6 FB
11	00-59 5F-A2 A8-F9
1E	00-9B A0-F9
1F	00-15 18-1D 20-45 48-4D 50-57 59 5B 5D 5F-7D 80-B4 B6-C4 C6-D3 D6-DB DD-EF F2-F4 F6-FE
20	00-2E 30-46 6A-70 74-8E A0-AB D0-E1
21	00-38 53-82 90-EA
22	00-F1
23	00 02-7A
24	00-24 40-4A 60-EA
25	00-95 A0-EF
26	00-13 1A-6F
27	01-04 06-09 0C-27 29-4B 4D 4F-52 56 58- 5E 61-67 76-94 98-AF B1-BE
30	00-37 3F 41-94 99-9E A1-FE
31	05-2C 31-8E 90-9F
32	00-1C 20-43 60-7B 7F-B0 C0-CB D0-FE
33	00-76 7B-DD E0-FE
4E-9F	4E00-9FA5
AC-D7	AC00-D7A3
E0-F8	E000-F8F8
F9-FA	F900-FA2D
FB	00-06 13-17 1E-36 38-3C 3E 40-41 43-44 46-B1 D3-FF
FC	00-FF
FD	00-3F 50-8F 92-C7 F0-FB
FE	20-23 30-44 49-52 54-66 68-6B 70-72 74 76-FC FF
FF	01-5E 61-BE C2-C7 CA-CF D2-D7 DA-DC E0-E6 E8-EE FD

A.3.2 299 BMP FIRST EDITION

The collection number and collection name 299 BMP FIRST EDITION have been reserved to identify the fixed collection comprising all of the coded characters that were in the BMP in the First Edition of ISO/IEC 10646-1. This collection is not now in conformity with this International Standard.

NOTE - The specification of collection 299 BMP FIRST EDITION consisted of the specification of collection 301 BMP-AMD.7 except for the replacement of the corresponding entries in the list above with the entries shown below:

rows positions

05	31-56 59-5F 61-87 89 B0-B9 BB-C3 D0-EA F0-F4
0F	[no positions]
1E	00-9A A0-F9
20	00-2E 30-46 6A-70 74-8E A0-AA D0-E1
AC-D7	[no positions]
34-4D	3400-4DFF

and by including an additional entry:
for the code position ranges of three collections (57, 58, 59) of coded characters which have been deleted from this International Standard since the First Edition of IO/IEC 10646-1.

A.3.3 302 BMP SECOND EDITION

The fixed collection 302 BMP SECOND EDITION comprises only those coded characters that are in the BMP in the Second Edition of ISO/IEC 10646-1. The repertoire of this collection is not subject to change if new characters are added to the BMP by any subsequent amendments.

302 BMP SECOND EDITION is specified by the following ranges of code positions as indicated for each row or contiguous series of rows.

<u>Rows</u>	<u>Positions (cells)</u>
00	20-7E A0-FF
01	00-FF
02	00-1F 22-33 50-AD B0-EE
03	00-4E 60-62 74-75 7A 7E 84-8A 8C 8E-A1 A3-CE D0-D7 DA-F3
04	00-86 88-89 8C-C4 C7-C8 CB-CC D0-F5 F8-F9
05	31-56 59-5F 61-87 89-8A 91-A1 A3-B9 BB- C4 D0-EA F0-F4
06	0C 1B 1F 21-3A 40-55 60-6D 70-ED F0-FE
07	00-0D 0F-2C 30-4A 80-B0
09	01-03 05-39 3C-4D 50-54 58-70 81-83 85- 8C 8F-90 93-A8 AA-B0 B2 B6-B9 BC BE- C4 C7-C8 CB-CD D7 DC-DD DF-E3 E6-FA
0A	02 05-0A 0F-10 13-28 2A-30 32-33 35-36 38-39 3C 3E-42 47-48 4B-4D 59-5C 5E 66- 74 81-83 85-8B 8D 8F-91 93-A8 AA-B0 B2- B3 B5-B9 BC-C5 C7-C9 CB-CD D0 E0 E6- EF
0B	01-03 05-0C 0F-10 13-28 2A-30 32-33 36-39 3C-43 47-48 4B-4D 56-57 5C-5D 5F-61 66- 70 82-83 85-8A 8E-90 92-95 99-9A 9C 9E- 9F A3-A4 A8-AA AE-B5 B7-B9 BE-C2 C6- C8 CA-CD D7 E7-F2
0C	01-03 05-0C 0E-10 12-28 2A-33 35-39 3E- 44 46-48 4A-4D 55-56 60-61 66-6F 82-83 85-8C 8E-90 92-A8 AA-B3 B5-B9 BE-C4 C6-C8 CA-CD D5-D6 DE E0-E1 E6-EF
0D	02-03 05-0C 0E-10 12-28 2A-39 3E-43 46- 48 4A-4D 57 60-61 66-6F 82-83 85-96 9A- B1 B3-BB BD C0-C6 CA CF-D4 D6 D8-DF F2-F4
0E	01-3A 3F-5B 81-82 84 87-88 8A 8D 94-97 99-9F A1-A3 A5 A7 AA-AB AD-B9 BB-BD C0-C4 C6 C8-CD D0-D9 DC-DD
0F	00-47 49-6A 71-8B 90-97 99-BC BE-CC CF

10	00-21 23-27 29-2A 2C-32 36-39 40-59 A0-C5 D0-F6 FB
11	00-59 5F-A2 A8-F9
12	00-06 08-46 48 4A-4D 50-56 58 5A-5D 60-86 88 8A-8D 90-AE B0 B2-B5 B8-BE C0 C2-C5 C8-CE D0-D6 D8-EE F0-FF
13	00-0E 10 12-15 18-1E 20-46 48-5A 61-7C A0-F4
14-15	1401-15FF
16	00-76 80-9C A0-F0
17	80-DC E0-E9
18	00-0E 10-19 20-77 80-A9
1E	00-9B A0-F9
1F	00-15 18-1D 20-45 48-4D 50-57 59 5B 5D 5F-7D 80-B4 B6-C4 C6-D3 D6-DB DD-EF F2-F4 F6-FE
20	00-46 48-4D 6A-70 74-8E A0-AF D0-E3
21	00-3A 53-83 90-F3
22	00-F1
23	00-7B 7D-9A
24	00-26 40-4A 60-EA
25	00-95 A0-F7
26	00-13 19-71
27	01-04 06-09 0C-27 29-4B 4D 4F-52 56 58-5E 61-67 76-94 98-AF B1-BE
28	00-FF
2E	80-99 9B-F3
2F	00-D5 F0-FB
30	00-3A 3E-3F 41-94 99-9E A1-FE
31	05-2C 31-8E 90-B7
32	00-1C 20-43 60-7B 7F-B0 C0-CB D0-FE
33	00-76 7B-DD E0-FE
34-4D	3400-4DB5
4E-9F	4E00-9FA5
A0-A3	A000-A3FF
A4	00-8C 90-A1 A4-B3 B5-C0 C2-C4 C6
AC-D7	AC00-D7A3
E0-F8	E000-F8FF
F9-FA	F900-FA2D
FB	00-06 13-17 1D-36 38-3C 3E 40-41 43-44 46-B1 D3-FF
FC	00-FF
FD	00-3F 50-8F 92-C7 F0-FB
FE	20-23 30-44 49-52 54-66 68-6B 70-72 74 76-FC FF
FF	01-5E 61-BE C2-C7 CA-CF D2-D7 DA-DC E0-E6 E8-EE F9-FD

A.4 Other collections within the BMP

The collections specified within this clause are entirely within Plane 00.

NOTE – The acronym MES indicates Multilingual European Subset.

A.4.1 281 MES-1

281 MES-1 is specified by the following ranges of code positions as indicated for each row.

Rows Positions (cells)

00	20-7E A0-FF
01	00-13 16-2B 2E-4D 50-7E
02	C7 D8-DB DD
20	15 18-19 1C-1D AC
21	22 26 5B-5E 90-93
26	6A

A.4.2 282 MES-2

282 MES-2 is specified by the following ranges of code positions as indicated for each row.

Rows Positions (cells)

00	20-7E A0-FF
01	00-7F 8F 92 B7 DE-EF FA-FF
02	18-1B 1E-1F 59 7C 92 BB-BD C6-C7 C9 D8-DD EE
03	74-75 7A 7E 84-8A 8C 8E-A1 A3-CE D7 DA-E1
04	00-5F 90-C4 C7-C8 CB-CC D0-EB EE-F5 F8-F9
1E	02-03 0A-0B 1E-1F 40-41 56-57 60-61 6A-6B 80-85 9B F2-F3
1F	00-15 18-1D 20-45 48-4D 50-57 59 5B 5D 5F-7D 80-B4 B6-C4 C6-D3 D6-DB DD-EF F2-F4 F6-FE
20	13-15 17-1E 20-22 26 30 32-33 39-3A 3C 3E 44 4A 7F 82 A3-A4 A7 AC AF
21	05 16 22 26 5B-5E 90-95 A8
22	00 02-03 06 08-09 0F 11-12 19-1A 1E-1F 27-2B 48 59 60-61 64-65 82-83 95 97
23	02 10 20-21 29-2A
25	00 02 0C 10 14 18 1C 24 2C 34 3C 50-6C 80 84 88 8C 90-93 A0 AC B2 BA BC C4 CA-CB D8-D9
26	3A-3C 40 42 60 63 65-66 6A-6B
FB	01-02
FF	FD

A.4.3 283 MODERN EUROPEAN SCRIPTS

283 MODERN EUROPEAN SCRIPTS is specified by the following collections:

Collection number and name

1	BASIC LATIN
2	LATIN-1 SUPPLEMENT
3	LATIN EXTENDED-A
4	LATIN EXTENDED-B
5	IPA EXTENSIONS
6	SPACING MODIFIER LETTERS
7	COMBINING DIACRITICAL MARKS
8	BASIC GREEK
9	GREEK SYMBOLS AND COPTIC
10	CYRILLIC
11	ARMENIAN
27	BASIC GEORGIAN
30	LATIN EXTENDED ADDITIONAL
31	GREEK EXTENDED
32	GENERAL PUNCTUATION
33	SUPERSCRIPTS AND SUBSCRIPTS
34	CURRENCY SYMBOLS
35	COMBINING DIACRITICAL MARKS FOR SYMBOLS
36	LETTERLIKE SYMBOLS

37	NUMBER FORMS
38	ARROWS
39	MATHEMATICAL OPERATORS
40	MISCELLANEOUS TECHNICAL
42	OPTICAL CHARACTER RECOGNITION
44	BOX DRAWING
45	BLOCK ELEMENTS
46	GEOMETRIC SHAPES
47	MISCELLANEOUS SYMBOLS
65	COMBINING HALF MARKS
70	SPECIALS
92	CYRILLIC SUPPLEMENT
104	LTR ALPHABETIC PRESENTATION FORMS

A.5 Unicode collections

These collections correspond to Unicode 3.1 and 3.2. They include characters from the BMP as well as Supplementary Planes.

A.5.1 303 UNICODE 3.1

303 The fixed collection UNICODE 3.1 consists of collections from A.3 above and several ranges of code positions. The collection list is arranged by planes as follows.

Plane 0

Collection number and name

302 BMP SECOND EDITION

Row Positions (cells)

03 F4-F5

Plane 1

Row Positions (cells)

03	00-1E 20-23 30-4A
04	00-25 28-4D
D0	00-F5
D1	00-26 2A-DD
D4	00-54 56-9C 9E-9F A2 A5-A6 A9-AC AE-B9 BB BD-C0 C2-C3 C5-FF
D5	00-05 07-0A 0D-14 16-1C 1E-39 3B-3E 40-44 46 4A-50 52-FF
D6	00-A3 A8-FF
D7	00-C9 CE-FF

Plane 2

Row Positions (cells)

00-A6	0000-A6D6
F8-FA	F800-FA1D

Plane 0E

Row Positions (cells)

00 01 20-7F

Plane 0F

Row Positions (cells)

00-FF 0000-FFFF

Plane 10

Row Positions (cells)

00-FF 0000-FFFF

A.5.2 304 UNICODE 3.2

304 The fixed collection UNICODE 3.2 consists of fixed collections from A.5.1 above and several ranges of code positions arranged by planes as follows.

Plane 0-10

Collection number and name

303 UNICODE 3.1

Plane 0

Collection number and name

98	SUPPLEMENTAL ARROWS-A
99	SUPPLEMENTAL ARROWS-B
100	MISCELLANEOUS MATHEMATICAL SYMBOLS-B
101	SUPPLEMENTAL MATHEMATICAL OPERATORS
102	KATAKANA PHONETIC EXTENSIONS
103	VARIATION SELECTORS

Rows Positions (cells)

02	20
03	4F 63-6F D8-D9 F6
04	8A-8B C5-C6 C9-CA CD-CE
05	00-0F
06	6E-6F
07	B1
10	F7-F8
17	00-0C 0E-14 20-36 40-53 60-6C 6E-70 72-73
20	47 4E-52 57 5F-63 71 B0-B1 E4-EA
21	3D-4B F4-FF
22	F2-FF
23	7C 9B-CE
24	EB-FE
25	96-9F F8-FF
26	16-17 72-7D 80-89
27	68-75 D0-EB
30	3B-3D 95-96 9F-A0 FF
32	51-5F B1-BF
A4	A2-A3 B4 C1 C5
FA	30-6A
FE	45-46 73
FF	5F-60

A.5.3 305 UNICODE 4.0

305 The fixed collection UNICODE 4.0 consists of fixed collections from A.5.2 above and several ranges of code positions arranged by planes as follows.

Working Draft ISO/IEC 10646:2003

Plane 0-10

Collection number and name

304 UNICODE 3.2

Plane 0

Collection number and name

108 KHMER SYMBOLS
111 YIJING HEXAGRAM SYMBOLS

Rows Positions

02 21 34-36 AE-AF EF-FF
03 50-57 5D-5F F7-F8
06 00-03 0D-14 56-59 EE-EF FF
07 2D-2F 4D-4F
09 04 BD
0A 01 03 8C E1-E3 F1
0B 35 F3-FA
0C BC-BD
17 DD F0-F9
19 00-1C 20-2B 30-3B 40 44-4F 50-6D 70-74
1D 00-6A
20 53-54
21 3B
23 CF
24 FF

26 14-15 8A-91 A0-A1
2B 00-0D
32 1D-1E 50 7C-7D CC-CF
33 77-7A DE-DF FF
FA 70-E9
FD FD
FE 47-48

Plane 1

Collection number and name

1011 SHAVIAN

Rows Positions

00 00-0B 0D-26 28-3A 3C-3D 3F-4D 50-5D 80-FA
01 00-02 07-33 37-3F
03 80-9D 9F
04 26-27 4E-4F 80-9D A0-A9
08 00-05 08 0A-35 37-38 3C 3F
D3 00-56
D4 C1

Plane E

Collection number and name

3003 VARIATION SELECTORS SUPPLEMENT

Annex B (normative)

List of combining characters

B.1 List of all combining characters

The characters in the subset collections COMBINING DIACRITICAL MARKS (0300 to 036F), COMBINING DIACRITICAL MARKS FOR SYMBOLS (20D0 to 20FF), and COMBINING HALF MARKS (FE20 to FE2F) are combining characters. In addition, the following characters are combining characters.

0483	COMBINING CYRILLIC TITLO	05B6	HEBREW POINT SEGOL
0484	COMBINING CYRILLIC PALATALIZATION	05B7	HEBREW POINT PATAH
0485	COMBINING CYRILLIC DASIA PNEUMATA	05B8	HEBREW POINT QAMATS
0486	COMBINING CYRILLIC PSILI PNEUMATA	05B9	HEBREW POINT HOLAM
0488	COMBINING CYRILLIC HUNDRED THOUSANDS SIGN	05BB	HEBREW POINT QUBUTS
		05BC	HEBREW POINT DAGESH OR MAPIQ
0489	COMBINING CYRILLIC MILLIONS SIGN	05BD	HEBREW POINT METEG
0591	HEBREW ACCENT ETNAHTA	05BF	HEBREW POINT RAFE
0592	HEBREW ACCENT SEGOL	05C1	HEBREW POINT SHIN DOT
0593	HEBREW ACCENT SHALSHELET	05C2	HEBREW POINT SIN DOT
0594	HEBREW ACCENT ZAQEF QATAN	05C4	HEBREW MARK UPPER DOT
0595	HEBREW ACCENT ZAQEF GADOL	0610	ARABIC SIGN SALLALLAHOU ALAYHE WASALLAM
0596	HEBREW ACCENT TIPEHA	0611	ARABIC SIGN ALAYHE ASSALAM
0597	HEBREW ACCENT REVIA	0612	ARABIC SIGN RAHMATULLAH ALAYHE
0598	HEBREW ACCENT ZARQA	0613	ARABIC SIGN RADI ALLAHOU ANHU
0599	HEBREW ACCENT PASHTA	0614	ARABIC SIGN TAKHALLUS
059A	HEBREW ACCENT YETIV	064B	ARABIC FATHATAN
059B	HEBREW ACCENT TEVIR	064C	ARABIC DAMMATAN
059C	HEBREW ACCENT GERESH	064D	ARABIC KASRATAN
059D	HEBREW ACCENT GERESH MUQDAM	064E	ARABIC FATHA
059E	HEBREW ACCENT GERSHAYIM	064F	ARABIC DAMMA
059F	HEBREW ACCENT QARNEY PARA	0650	ARABIC KASRA
05A0	HEBREW ACCENT TELISHA GEDOLA	0651	ARABIC SHADDA
05A1	HEBREW ACCENT PAZER	0652	ARABIC SUKUN
05A3	HEBREW ACCENT MUNAH	0653	ARABIC MADDAAH ABOVE
05A4	HEBREW ACCENT MAHAPAKH	0654	ARABIC HAMZA ABOVE
05A5	HEBREW ACCENT MERKHA	0655	ARABIC HAMZA BELOW
05A6	HEBREW ACCENT MERKHA KEFULA	0656	ARABIC SUBSCRIPT ALEF
05A7	HEBREW ACCENT DARGA	0657	ARABIC INVERTED DAMMA
05A8	HEBREW ACCENT QADMA	0658	ARABIC NOON-GHUNNA
05A9	HEBREW ACCENT TELISHA QETANA	0659	ARABIC SMALL HIGH TAH
05AA	HEBREW ACCENT YERAH BEN YOMO	0670	ARABIC LETTER SUPERSCRIPIT ALEF
05AB	HEBREW ACCENT OLE	06D7	ARABIC SMALL HIGH LIGATURE QAF WITH LAM WITH ALEF MAKSURA
05AC	HEBREW ACCENT ILUY	06D8	ARABIC SMALL HIGH MEEM INITIAL FORM
05AD	HEBREW ACCENT DEHI	06D9	ARABIC SMALL HIGH LAM ALEF
05AE	HEBREW ACCENT ZINOR	06DA	ARABIC SMALL HIGH JEEM
05AF	HEBREW MARK MASORA CIRCLE	06DB	ARABIC SMALL HIGH THREE DOTS
05B0	HEBREW POINT SHEVA	06DC	ARABIC SMALL HIGH SEEN
05B1	HEBREW POINT HATAF SEGOL	06DE	ARABIC START OF RUB EL HIZB
05B2	HEBREW POINT HATAF PATAH	06DF	ARABIC SMALL HIGH ROUNDED ZERO
05B3	HEBREW POINT HATAF QAMATS	06E0	ARABIC SMALL HIGH UPRIGHT RECTANGULAR ZERO
05B4	HEBREW POINT HIRIQ	06E1	ARABIC SMALL HIGH DOTLESS HEAD OF KHAH
05B5	HEBREW POINT TSERE	06E2	ARABIC SMALL HIGH MEEM ISOLATED FORM
		06E3	ARABIC SMALL LOW SEEN
		06E4	ARABIC SMALL HIGH MADDA
		06E7	ARABIC SMALL HIGH YEH
		06E8	ARABIC SMALL HIGH NOON
		06EA	ARABIC EMPTY CENTRE LOW STOP
		06EB	ARABIC EMPTY CENTRE HIGH STOP

06EC	ARABIC ROUNDED HIGH STOP WITH FILLED CENTRE	0953	DEVANAGARI GRAVE ACCENT
06ED	ARABIC SMALL LOW MEEM	0954	DEVANAGARI ACUTE ACCENT
0711	SYRIAC LETTER SUPERScript ALAPH	0962	DEVANAGARI VOWEL SIGN VOCALIC L
0730	SYRIAC PTHAHA ABOVE	0963	DEVANAGARI VOWEL SIGN VOCALIC LL
0731	SYRIAC PTHAHA BELOW	0981	BENGALI SIGN CANDRABINDU
0732	SYRIAC PTHAHA DOTTED	0982	BENGALI SIGN ANUSVARA
0733	SYRIAC ZQAPHA ABOVE	0983	BENGALI SIGN VISARGA
0734	SYRIAC ZQAPHA BELOW	09BC	BENGALI SIGN NUKTA
0735	SYRIAC ZQAPHA DOTTED	09BE	BENGALI VOWEL SIGN AA
0736	SYRIAC RBASA ABOVE	09BF	BENGALI VOWEL SIGN I
0737	SYRIAC RBASA BELOW	09C0	BENGALI VOWEL SIGN II
0738	SYRIAC DOTTED ZLAMA HORIZONTAL	09C1	BENGALI VOWEL SIGN U
0739	SYRIAC DOTTED ZLAMA ANGULAR	09C2	BENGALI VOWEL SIGN UU
073A	SYRIAC HBASA ABOVE	09C3	BENGALI VOWEL SIGN VOCALIC R
073B	SYRIAC HBASA BELOW	09C4	BENGALI VOWEL SIGN VOCALIC RR
073C	SYRIAC HBASA-ESASA DOTTED	09C7	BENGALI VOWEL SIGN E
073D	SYRIAC ESASA ABOVE	09C8	BENGALI VOWEL SIGN AI
073E	SYRIAC ESASA BELOW	09CB	BENGALI VOWEL SIGN O
073F	SYRIAC RWAHA	09CC	BENGALI VOWEL SIGN AU
0740	SYRIAC FEMININE DOT	09CD	BENGALI SIGN VIRAMA
0741	SYRIAC QUSHSHAYA	09D7	BENGALI AU LENGTH MARK
0742	SYRIAC RUKKAKHA	09E2	BENGALI VOWEL SIGN VOCALIC L
0743	SYRIAC TWO VERTICAL DOTS ABOVE	09E3	BENGALI VOWEL SIGN VOCALIC LL
0744	SYRIAC TWO VERTICAL DOTS BELOW	0A01	GURMUKHI SIGN ADAK BINDI
0745	SYRIAC THREE DOTS ABOVE	0A02	GURMUKHI SIGN BINDI
0746	SYRIAC THREE DOTS BELOW	0A03	GURMUKHI SIGN VISARGA
0747	SYRIAC OBLIQUE LINE ABOVE	0A3C	GURMUKHI SIGN NUKTA
0748	SYRIAC OBLIQUE LINE BELOW	0A3E	GURMUKHI VOWEL SIGN AA
0749	SYRIAC MUSIC	0A3F	GURMUKHI VOWEL SIGN I
074A	SYRIAC BARREKH	0A40	GURMUKHI VOWEL SIGN II
07A6	THAANA ABAFILI	0A41	GURMUKHI VOWEL SIGN U
07A7	THAANA AABAFILE	0A42	GURMUKHI VOWEL SIGN UU
07A8	THAANA IBIFILI	0A47	GURMUKHI VOWEL SIGN EE
07A9	THAANA EEBEFILI	0A48	GURMUKHI VOWEL SIGN AI
07AA	THAANA UBUFILI	0A4B	GURMUKHI VOWEL SIGN OO
07AB	THAANA OBOOFILI	0A4C	GURMUKHI VOWEL SIGN AU
07AC	THAANA EBEFILI	0A4D	GURMUKHI SIGN VIRAMA
07AD	THAANA EYBEFILI	0A70	GURMUKHI TIPPI
07AE	THAANA OBOFILI	0A71	GURMUKHI ADDAK
07AF	THAANA OABOAFILI	0A81	GUJARATI SIGN CANDRABINDU
07B0	THAANA SUKUN	0A82	GUJARATI SIGN ANUSVARA
0901	DEVANAGARI SIGN CANDRABINDU	0A83	GUJARATI SIGN VISARGA
0902	DEVANAGARI SIGN ANUSVARA	0ABC	GUJARATI SIGN NUKTA
0903	DEVANAGARI SIGN VISARGA	0ABE	GUJARATI VOWEL SIGN AA
093C	DEVANAGARI SIGN NUKTA	0ABF	GUJARATI VOWEL SIGN I
093E	DEVANAGARI VOWEL SIGN AA	0AC0	GUJARATI VOWEL SIGN II
093F	DEVANAGARI VOWEL SIGN I	0AC1	GUJARATI VOWEL SIGN U
0940	DEVANAGARI VOWEL SIGN II	0AC2	GUJARATI VOWEL SIGN UU
0941	DEVANAGARI VOWEL SIGN U	0AC3	GUJARATI VOWEL SIGN VOCALIC R
0942	DEVANAGARI VOWEL SIGN UU	0AC4	GUJARATI VOWEL SIGN VOCALIC RR
0943	DEVANAGARI VOWEL SIGN VOCALIC R	0AC5	GUJARATI VOWEL SIGN CANDRA E
0944	DEVANAGARI VOWEL SIGN VOCALIC RR	0AC7	GUJARATI VOWEL SIGN E
0945	DEVANAGARI VOWEL SIGN CANDRA E	0AC8	GUJARATI VOWEL SIGN AI
0946	DEVANAGARI VOWEL SIGN SHORT E	0AC9	GUJARATI VOWEL SIGN CANDRA O
0947	DEVANAGARI VOWEL SIGN E	0ACB	GUJARATI VOWEL SIGN O
0948	DEVANAGARI VOWEL SIGN AI	0ACC	GUJARATI VOWEL SIGN AU
0949	DEVANAGARI VOWEL SIGN CANDRA O	0ACD	GUJARATI SIGN VIRAMA
094A	DEVANAGARI VOWEL SIGN SHORT O	0AE2	GUJARATI VOWEL SIGN VOCALIC L
094B	DEVANAGARI VOWEL SIGN O	0AE3	GUJARATI VOWEL SIGN VOCALIC LL
094C	DEVANAGARI VOWEL SIGN AU	0B01	ORIYA SIGN CANDRABINDU
094D	DEVANAGARI SIGN VIRAMA	0B02	ORIYA SIGN ANUSVARA
0951	DEVANAGARI STRESS SIGN UDATTA	0B03	ORIYA SIGN VISARGA
0952	DEVANAGARI STRESS SIGN ANUDATTA	0B3C	ORIYA SIGN NUKTA
		0B3E	ORIYA VOWEL SIGN AA

0B3F	ORIYA VOWEL SIGN I	0D02	MALAYALAM SIGN ANUSVARA
0B40	ORIYA VOWEL SIGN II	0D03	MALAYALAM SIGN VISARGA
0B41	ORIYA VOWEL SIGN U	0D3E	MALAYALAM VOWEL SIGN AA
0B42	ORIYA VOWEL SIGN UU	0D3F	MALAYALAM VOWEL SIGN I
0B43	ORIYA VOWEL SIGN VOCALIC R	0D40	MALAYALAM VOWEL SIGN II
0B47	ORIYA VOWEL SIGN E	0D41	MALAYALAM VOWEL SIGN U
0B48	ORIYA VOWEL SIGN AI	0D42	MALAYALAM VOWEL SIGN UU
0B4B	ORIYA VOWEL SIGN O	0D43	MALAYALAM VOWEL SIGN VOCALIC R
0B4C	ORIYA VOWEL SIGN AU	0D46	MALAYALAM VOWEL SIGN E
0B4D	ORIYA SIGN VIRAMA	0D47	MALAYALAM VOWEL SIGN EE
0B56	ORIYA AI LENGTH MARK	0D48	MALAYALAM VOWEL SIGN AI
0B57	ORIYA AU LENGTH MARK	0D4A	MALAYALAM VOWEL SIGN O
0B82	TAMIL SIGN ANUSVARA	0D4B	MALAYALAM VOWEL SIGN OO
0BBE	TAMIL VOWEL SIGN AA	0D4C	MALAYALAM VOWEL SIGN AU
0BBF	TAMIL VOWEL SIGN I	0D4D	MALAYALAM SIGN VIRAMA
0BC0	TAMIL VOWEL SIGN II	0D57	MALAYALAM AU LENGTH MARK
0BC1	TAMIL VOWEL SIGN U	0D82	SINHALA SIGN ANUSVARAYA
0BC2	TAMIL VOWEL SIGN UU	0D83	SINHALA SIGN VISARGAYA
0BC6	TAMIL VOWEL SIGN E	0DCA	SINHALA SIGN AL-LAKUNA
0BC7	TAMIL VOWEL SIGN EE	0DCF	SINHALA VOWEL SIGN AELA-PILLA
0BC8	TAMIL VOWEL SIGN AI	0DD0	SINHALA VOWEL SIGN KETTI AEDA-PILLA
0BCA	TAMIL VOWEL SIGN O	0DD1	SINHALA VOWEL SIGN DIGA AEDA-PILLA
0BCB	TAMIL VOWEL SIGN OO	0DD2	SINHALA VOWEL SIGN KETTI IS-PILLA
0BCC	TAMIL VOWEL SIGN AU	0DD3	SINHALA VOWEL SIGN DIGA IS-PILLA
0BCD	TAMIL SIGN VIRAMA	0DD4	SINHALA VOWEL SIGN KETTI PAA-PILLA
0BD7	TAMIL AU LENGTH MARK	0DD6	SINHALA VOWEL SIGN DIGA PAA-PILLA
0C01	TELUGU SIGN CANDRABINDU	0DD8	SINHALA VOWEL SIGN GAETTA-PILLA
0C02	TELUGU SIGN ANUSVARA	0DD9	SINHALA VOWEL SIGN KOMBUVA
0C03	TELUGU SIGN VISARGA	0DDA	SINHALA VOWEL SIGN DIGA KOMBUVA
0C3E	TELUGU VOWEL SIGN AA	0ddb	SINHALA VOWEL SIGN KOMBU DEKA
0C3F	TELUGU VOWEL SIGN I	0DDC	SINHALA VOWEL SIGN KOMBUVA HAA AELA-PILLA
0C40	TELUGU VOWEL SIGN II	0DDD	SINHALA VOWEL SIGN KOMBUVA HAA DIGA AELA-PILLA
0C41	TELUGU VOWEL SIGN U	0DDE	SINHALA VOWEL SIGN KOMBUVA HAA GAYANUKITTA
0C42	TELUGU VOWEL SIGN UU	0DDF	SINHALA VOWEL SIGN GAYANUKITTA
0C43	TELUGU VOWEL SIGN VOCALIC R	0DF2	SINHALA VOWEL SIGN DIGA GAETTA-PILLA
0C44	TELUGU VOWEL SIGN VOCALIC RR	0DF3	SINHALA VOWEL SIGN DIGA GAYANUKITTA
0C46	TELUGU VOWEL SIGN E	0E31	THAI CHARACTER MAI HAN-AKAT
0C47	TELUGU VOWEL SIGN EE	0E34	THAI CHARACTER SARA I
0C48	TELUGU VOWEL SIGN AI	0E35	THAI CHARACTER SARA II
0C4A	TELUGU VOWEL SIGN O	0E36	THAI CHARACTER SARA UE
0C4B	TELUGU VOWEL SIGN OO	0E37	THAI CHARACTER SARA UEE
0C4C	TELUGU VOWEL SIGN AU	0E38	THAI CHARACTER SARA U
0C4D	TELUGU SIGN VIRAMA	0E39	THAI CHARACTER SARA UU
0C55	TELUGU LENGTH MARK	0E3A	THAI CHARACTER PHINTHU
0C56	TELUGU AI LENGTH MARK	0E47	THAI CHARACTER MAITAIKHU
0C82	KANNADA SIGN ANUSVARA	0E48	THAI CHARACTER MAI EK
0C83	KANNADA SIGN VISARGA	0E49	THAI CHARACTER MAI THO
0CBC	KANNADA SIGN NUKTA	0E4A	THAI CHARACTER MAI TRI
0CBE	KANNADA VOWEL SIGN AA	0E4B	THAI CHARACTER MAI CHATTAWA
0CBF	KANNADA VOWEL SIGN I	0E4C	THAI CHARACTER THANTHAKHAT
0CC0	KANNADA VOWEL SIGN II	0E4D	THAI CHARACTER NIKHAHIT
0CC1	KANNADA VOWEL SIGN U	0E4E	THAI CHARACTER YAMAKKAN
0CC2	KANNADA VOWEL SIGN UU	0EB1	LAO VOWEL SIGN MAI KAN
0CC3	KANNADA VOWEL SIGN VOCALIC R	0EB4	LAO VOWEL SIGN I
0CC4	KANNADA VOWEL SIGN VOCALIC RR	0EB5	LAO VOWEL SIGN II
0CC6	KANNADA VOWEL SIGN E	0EB6	LAO VOWEL SIGN Y
0CC7	KANNADA VOWEL SIGN EE	0EB7	LAO VOWEL SIGN YY
0CC8	KANNADA VOWEL SIGN AI	0EB8	LAO VOWEL SIGN U
0CCA	KANNADA VOWEL SIGN O	0EB9	LAO VOWEL SIGN UU
0CCB	KANNADA VOWEL SIGN OO	0EBB	LAO VOWEL SIGN MAI KON
0CCC	KANNADA VOWEL SIGN AU	0EBC	LAO SEMIVOWEL SIGN LO
0CCD	KANNADA SIGN VIRAMA	0EC8	LAO TONE MAI EK
0CD5	KANNADA LENGTH MARK		
0CD6	KANNADA AI LENGTH MARK		

0EC9	LAO TONE MAI THO	0FAF	TIBETAN SUBJOINED LETTER ZA
0ECA	LAO TONE MAI TI	0FB0	TIBETAN SUBJOINED LETTER -A
0ECB	LAO TONE MAI CATAWA	0FB1	TIBETAN SUBJOINED LETTER YA
0ECC	LAO CANCELLATION MARK	0FB2	TIBETAN SUBJOINED LETTER RA
0ECD	LAO NIGGAHITA	0FB3	TIBETAN SUBJOINED LETTER LA
0F18	TIBETAN ASTROLOGICAL SIGN -KHYUD PA	0FB4	TIBETAN SUBJOINED LETTER SHA
0F19	TIBETAN ASTROLOGICAL SIGN SDONG TSHUGS	0FB5	TIBETAN SUBJOINED LETTER SSA
0F35	TIBETAN MARK NGAS BZUNG NYI ZLA	0FB6	TIBETAN SUBJOINED LETTER SA
0F37	TIBETAN MARK NGAS BZUNG SGOR RTAGS	0FB7	TIBETAN SUBJOINED LETTER HA
0F39	TIBETAN MARK TSA -PHRU	0FB8	TIBETAN SUBJOINED LETTER A
0F3E	TIBETAN SIGN YAR TSHES	0FB9	TIBETAN SUBJOINED LETTER KSSA
0F3F	TIBETAN SIGN MAR TSHES	0FBA	TIBETAN SUBJOINED LETTER FIXED-FORM WA
0F71	TIBETAN VOWEL SIGN AA	0FBB	TIBETAN SUBJOINED LETTER FIXED-FORM YA
0F72	TIBETAN VOWEL SIGN I	0FBC	TIBETAN SUBJOINED LETTER FIXED-FORM RA
0F73	TIBETAN VOWEL SIGN II	0FC6	TIBETAN SYMBOL PADMA GDAN
0F74	TIBETAN VOWEL SIGN U	102C	MYANMAR VOWEL SIGN AA
0F75	TIBETAN VOWEL SIGN UU	102D	MYANMAR VOWEL SIGN I
0F76	TIBETAN VOWEL SIGN VOCALIC R	102E	MYANMAR VOWEL SIGN II
0F77	TIBETAN VOWEL SIGN VOCALIC RR	102F	MYANMAR VOWEL SIGN U
0F78	TIBETAN VOWEL SIGN VOCALIC L	1030	MYANMAR VOWEL SIGN UU
0F79	TIBETAN VOWEL SIGN VOCALIC LL	1031	MYANMAR VOWEL SIGN E
0F7A	TIBETAN VOWEL SIGN E	1032	MYANMAR VOWEL SIGN AI
0F7B	TIBETAN VOWEL SIGN EE	1036	MYANMAR SIGN ANUSVARA
0F7C	TIBETAN VOWEL SIGN O	1037	MYANMAR SIGN DOT BELOW
0F7D	TIBETAN VOWEL SIGN OO	1038	MYANMAR SIGN VISARGA
0F7E	TIBETAN SIGN RJES SU NGA RO	1039	MYANMAR SIGN VIRAMA
0F7F	TIBETAN SIGN RNAM BCAD	1056	MYANMAR VOWEL SIGN VOCALIC R
0F80	TIBETAN VOWEL SIGN REVERSED I	1057	MYANMAR VOWEL SIGN VOCALIC RR
0F81	TIBETAN VOWEL SIGN REVERSED II	1058	MYANMAR VOWEL SIGN VOCALIC L
0F82	TIBETAN SIGN NYI ZLA NAA DA	1059	MYANMAR VOWEL SIGN VOCALIC LL
0F83	TIBETAN SIGN SNA LDAN	1712	TAGALOG VOWEL SIGN I
0F84	TIBETAN MARK HALANTA	1713	TAGALOG VOWEL SIGN U
0F86	TIBETAN MARK LCI RTAGS	1714	TAGALOG VIRAMA
0F87	TIBETAN MARK YANG RTAGS	1732	HANUNOO VOWEL SIGN I
0F90	TIBETAN SUBJOINED LETTER KA	1733	HANUNOO VOWEL SIGN U
0F91	TIBETAN SUBJOINED LETTER KHA	1734	HANUNOO PAMUDPOD
0F92	TIBETAN SUBJOINED LETTER GA	1752	BUHID VOWEL SIGN I
0F93	TIBETAN SUBJOINED LETTER GHA	1753	BUHID VOWEL SIGN U
0F94	TIBETAN SUBJOINED LETTER NGA	1772	TAGBANWA VOWEL SIGN I
0F95	TIBETAN SUBJOINED LETTER CA	1773	TAGBANWA VOWEL SIGN U
0F96	TIBETAN SUBJOINED LETTER CHA	17B6	KHMER VOWEL SIGN AA
0F97	TIBETAN SUBJOINED LETTER JA	17B7	KHMER VOWEL SIGN I
0F99	TIBETAN SUBJOINED LETTER NYA	17B8	KHMER VOWEL SIGN II
0F9A	TIBETAN SUBJOINED LETTER TTA	17B9	KHMER VOWEL SIGN Y
0F9B	TIBETAN SUBJOINED LETTER TTHA	17BA	KHMER VOWEL SIGN YY
0F9C	TIBETAN SUBJOINED LETTER DDA	17BB	KHMER VOWEL SIGN U
0F9D	TIBETAN SUBJOINED LETTER DDHA	17BC	KHMER VOWEL SIGN UU
0F9E	TIBETAN SUBJOINED LETTER NNA	17BD	KHMER VOWEL SIGN UA
0F9F	TIBETAN SUBJOINED LETTER TA	17BE	KHMER VOWEL SIGN OE
0FA0	TIBETAN SUBJOINED LETTER THA	17BF	KHMER VOWEL SIGN YA
0FA1	TIBETAN SUBJOINED LETTER DA	17C0	KHMER VOWEL SIGN IE
0FA2	TIBETAN SUBJOINED LETTER DHA	17C1	KHMER VOWEL SIGN E
0FA3	TIBETAN SUBJOINED LETTER NA	17C2	KHMER VOWEL SIGN AE
0FA4	TIBETAN SUBJOINED LETTER PA	17C3	KHMER VOWEL SIGN AI
0FA5	TIBETAN SUBJOINED LETTER PHA	17C4	KHMER VOWEL SIGN OO
0FA6	TIBETAN SUBJOINED LETTER BA	17C5	KHMER VOWEL SIGN AU
0FA7	TIBETAN SUBJOINED LETTER BHA	17C6	KHMER SIGN NIKAHIT
0FA8	TIBETAN SUBJOINED LETTER MA	17C7	KHMER SIGN REAHMUK
0FA9	TIBETAN SUBJOINED LETTER TSA	17C8	KHMER SIGN YUUKALEAPINTU
0FAA	TIBETAN SUBJOINED LETTER TSHA	17C9	KHMER SIGN MUUSIKATOAN
0FAB	TIBETAN SUBJOINED LETTER DZA	17CA	KHMER SIGN TRIISAP
0FAC	TIBETAN SUBJOINED LETTER DZHA	17CB	KHMER SIGN BANTOC
0FAD	TIBETAN SUBJOINED LETTER WA	17CC	KHMER SIGN ROBAT
0FAE	TIBETAN SUBJOINED LETTER ZHA	17CD	KHMER SIGN TOANDAKHIAT

17CE	KHMER SIGN KAKABAT	1D166	MUSICAL SYMBOL COMBINING SPRECHGESANG STEM
17CF	KHMER SIGN AHSDA	1D167	MUSICAL SYMBOL COMBINING TREMOLO ONE
17D0	KHMER SIGN SAMYOK SANNYA	1D168	MUSICAL SYMBOL COMBINING TREMOLO TWO
17D1	KHMER SIGN VIRIAM	1D169	MUSICAL SYMBOL COMBINING TREMOLO THREE
17D2	KHMER SIGN COENG	1D16D	MUSICAL SYMBOL COMBINING AUGMENTATION DOT
17D3	KHMER SIGN BATHAMASAT	1D16E	MUSICAL SYMBOL COMBINING FLAG ONE
17DD	KHMER SIGN ATTHACAN	1D16F	MUSICAL SYMBOL COMBINING FLAG TWO
180B	MONGOLIAN FREE VARIATION SELECTOR ONE	1D170	MUSICAL SYMBOL COMBINING FLAG THREE
180C	MONGOLIAN FREE VARIATION SELECTOR TWO	1D171	MUSICAL SYMBOL COMBINING FLAG FOUR
180D	MONGOLIAN FREE VARIATION SELECTOR THREE	1D172	MUSICAL SYMBOL COMBINING FLAG FIVE
18A9	MONGOLIAN LETTER AG DAGALGA	1D17B	MUSICAL SYMBOL COMBINING ACCENT
1920	LIMBU VOWEL SIGN A	1D17C	MUSICAL SYMBOL COMBINING STACCATO
1921	LIMBU VOWEL SIGN I	1D17D	MUSICAL SYMBOL COMBINING TENUTO
1922	LIMBU VOWEL SIGN U	1D17E	MUSICAL SYMBOL COMBINING STACCATISSIMO
1923	LIMBU VOWEL SIGN EE	1D17F	MUSICAL SYMBOL COMBINING MARCATO
1924	LIMBU VOWEL SIGN AI	1D180	MUSICAL SYMBOL COMBINING MARCATO STACCATO
1925	LIMBU VOWEL SIGN OO	1D181	MUSICAL SYMBOL COMBINING ACCENT-STACCATO
1926	LIMBU VOWEL SIGN AU	1D182	MUSICAL SYMBOL COMBINING LOURE
1927	LIMBU VOWEL SIGN E	1D185	MUSICAL SYMBOL COMBINING DOIT
1928	LIMBU VOWEL SIGN O	1D186	MUSICAL SYMBOL COMBINING RIP
1929	LIMBU SUBJOINED LETTER YA	1D187	MUSICAL SYMBOL COMBINING FLIP
192A	LIMBU SUBJOINED LETTER RA	1D188	MUSICAL SYMBOL COMBINING SMEAR
192B	LIMBU SUBJOINED LETTER WA	1D189	MUSICAL SYMBOL COMBINING BEND
1930	LIMBU SMALL LETTER KA	1D18A	MUSICAL SYMBOL COMBINING DOUBLE TONGUE
1931	LIMBU SMALL LETTER NGA	1D18B	MUSICAL SYMBOL COMBINING TRIPLE TONGUE
1932	LIMBU SMALL LETTER ANUSVARA	1D1AA	MUSICAL SYMBOL COMBINING DOWN BOW
1933	LIMBU SMALL LETTER TA	1D1AB	MUSICAL SYMBOL COMBINING UP BOW
1934	LIMBU SMALL LETTER NA	1D1AC	MUSICAL SYMBOL COMBINING HARMONIC
1935	LIMBU SMALL LETTER PA	1D1AD	MUSICAL SYMBOL COMBINING SNAP PIZZICATO
1936	LIMBU SMALL LETTER MA		
1937	LIMBU SMALL LETTER RA		
1938	LIMBU SMALL LETTER LA		
1939	LIMBU SIGN MUKPHRENG		
193A	LIMBU SIGN KEMPHRENG		
193B	LIMBU SIGN SA-I		
302A	IDEOGRAPHIC LEVEL TONE MARK		
302B	IDEOGRAPHIC RISING TONE MARK		
302C	IDEOGRAPHIC DEPARTING TONE MARK		
302D	IDEOGRAPHIC ENTERING TONE MARK		
302E	HANGUL SINGLE DOT TONE MARK		
302F	HANGUL DOUBLE DOT TONE MARK		
3099	COMBINING KATAKANA-HIRAGANA VOICED SOUND MARK		
309A	COMBINING KATAKANA-HIRAGANA SEMI-VOICED SOUND MARK		
FB1E	HEBREW POINT JUDEO-SPANISH VARIKA		
FE00	VARIATION SELECTOR-1		
FE01	VARIATION SELECTOR-2		
FE02	VARIATION SELECTOR-3		
FE03	VARIATION SELECTOR-4		
FE04	VARIATION SELECTOR-5		
FE05	VARIATION SELECTOR-6		
FE06	VARIATION SELECTOR-7		
FE07	VARIATION SELECTOR-8		
FE08	VARIATION SELECTOR-9		
FE09	VARIATION SELECTOR-10		
FE0A	VARIATION SELECTOR-11		
FE0B	VARIATION SELECTOR-12		
FE0C	VARIATION SELECTOR-13		
FE0D	VARIATION SELECTOR-14		
FE0E	VARIATION SELECTOR-15		
FE0F	VARIATION SELECTOR-16		
1D165	MUSICAL SYMBOL COMBINING STEM		

B.2 List of combining and other characters not allowed in implementation level 2

The characters in the subset collections COMBINING DIACRITICAL MARKS (0300 to 036F), COMBINING DIACRITICAL MARKS FOR SYMBOLS (20D0 to 20FF), HANGUL JAMO (1100 to 11FF) and COMBINING HALF MARKS (FE20 to FE2F) are not allowed in implementation level 2. In addition, the following individual characters are also not allowed.

NOTE - This list is a subset of the list in clause B.1 except for HANGUL JAMO (see 25.1).

0483	COMBINING CYRILLIC TITLO	05A6	HEBREW ACCENT MERKHA KEFULA
0484	COMBINING CYRILLIC PALATALIZATION	05A7	HEBREW ACCENT DARGA
0485	COMBINING CYRILLIC DASIA PNEUMATA	05A8	HEBREW ACCENT QADMA
0486	COMBINING CYRILLIC PSILI PNEUMATA	05A9	HEBREW ACCENT TELISHA QETANA
0591	HEBREW ACCENT ETNAHTA	05AA	HEBREW ACCENT YERAH BEN YOMO
0592	HEBREW ACCENT SEGOL	05AB	HEBREW ACCENT OLE
0593	HEBREW ACCENT SHALSHELET	05AC	HEBREW ACCENT ILUY
0594	HEBREW ACCENT ZAQEF QATAN	05AD	HEBREW ACCENT DEHI
0595	HEBREW ACCENT ZAQEF GADOL	05AE	HEBREW ACCENT ZINOR
0596	HEBREW ACCENT TIPEHA	05AF	HEBREW MARK MASORA CIRCLE
0597	HEBREW ACCENT REVIA	05C4	HEBREW MARK UPPER DOT
0598	HEBREW ACCENT ZARQA	093C	DEVANAGARI SIGN NUKTA
0599	HEBREW ACCENT PASHTA	0953	DEVANAGARI GRAVE ACCENT
059A	HEBREW ACCENT YETIV	0954	DEVANAGARI ACUTE ACCENT
059B	HEBREW ACCENT TEVIR	09BC	BENGALI SIGN NUKTA
059C	HEBREW ACCENT GERESH	09D7	BENGALI AU LENGTH MARK
059D	HEBREW ACCENT GERESH MUQDAM	0A3C	GURMUKHI SIGN NUKTA
059E	HEBREW ACCENT GERSHAYIM	0A70	GURMUKHI TIPPI
059F	HEBREW ACCENT QARNEY PARA	0A71	GURMUKHI ADDAK
05A0	HEBREW ACCENT TELISHA GEDOLA	0ABC	GUJARATI SIGN NUKTA
05A1	HEBREW ACCENT PAZER	0B3C	ORIYA SIGN NUKTA
05A3	HEBREW ACCENT MUNAH	0B56	ORIYA AI LENGTH MARK
05A4	HEBREW ACCENT MAHAPAKH	0B57	ORIYA AU LENGTH MARK
05A5	HEBREW ACCENT MERKHA	0BD7	TAMIL AU LENGTH MARK
		0C55	TELUGU LENGTH MARK
		0C56	TELUGU AI LENGTH MARK
		0CD5	KANNADA LENGTH MARK
		0CD6	KANNADA AI LENGTH MARK
		0D57	MALAYALAM AU LENGTH MARK
		0F39	TIBETAN MARK TSA -PHRU
		302A	IDEOGRAPHIC LEVEL TONE MARK
		302B	IDEOGRAPHIC RISING TONE MARK
		302C	IDEOGRAPHIC DEPARTING TONE MARK
		302D	IDEOGRAPHIC ENTERING TONE MARK
		302E	HANGUL SINGLE DOT TONE MARK
		302F	HANGUL DOUBLE DOT TONE MARK
		3099	COMBINING KATAKANA-HIRAGANA VOICED SOUND MARK
		309A	COMBINING KATAKANA-HIRAGANA SEMI-VOICED SOUND MARK

Annex C (normative)

Transformation format for 16 planes of Group 00 (UTF-16)

UTF-16 provides a coded representation of over a million graphic characters of UCS-4 in a form that is compatible with the two-octet BMP form of UCS-2 (13.1). This permits the coexistence of those characters from UCS-4 within coded character data that is in accordance with UCS-2.

In UTF-16 each graphic character from the BMP repertoire retains its UCS-2 coded representation. In addition, the coded representation of any character from a single contiguous block of 16 Planes in Group 00 (1,048,576 code positions) consists of a pair of RC-elements (4.33), where each such RC-element corresponds to a cell in a single contiguous block of 8 Rows in the BMP (2048 code positions). These code positions are reserved for the use of this coded representation form, and shall not be allocated for any other purpose.

C.1 Specification of UTF-16

The specification of UTF-16 is as follows:

1. The high-half zone shall be the 4 rows D8 to DB of the BMP, i.e., the 1024 cells in the S-zone whose code positions are from D800 through DBFF.
2. The low-half zone shall be the 4 rows DC to DF of the BMP, i.e., the 1024 cells in the S-zone whose code positions are from DC00 through DFFF.
3. All cells in the high-half zone and the low-half zone shall be permanently reserved for the use of the UTF-16 coded representation form.
4. In UTF-16, any UCS character from the BMP shall be represented by its UCS-2 coded representation as specified by the body of this international standard.
5. In UTF-16, any UCS character whose UCS-4 coded representation is in the range 0001 0000 to 0010 FFFF shall be represented by a sequence of two RC-elements from the S-zone, of which the first is an RC-element from the high-half zone, and the second is an RC-element from the low-half zone.

The mapping between UCS-4 and UTF-16 for these characters shall be as shown in C.3; the reverse mapping is shown in C.4.

NOTE - The Unicode Standard, Version 3.0, defines the following forms of UTF-16.

- UTF-16: the ordering of octets (6.3) is not defined and signatures (Annex H) may appear;
- UTF-16BE: in the ordering of octets the more significant octet precedes the less significant octet, as specified in 6.2, and no signatures appear;
- UTF-16LE: in the ordering of octets the less significant octet precedes the more significant octet and no signatures appear.

C.2 Notation

1. All numbers are in hexadecimal notation.
2. Double-octet boundaries in the notations for UTF-16 are indicated with semicolons.
3. The symbol “%” indicates the modulo operation, e.g.: $7 \% 3 = 1$.
4. The symbol “/” indicates the integer division operation, e.g.: $7 / 3 = 2$.
5. Precedence is -
integer-division > modulo-operation >
integer-multiplication > integer-addition.

C.3 Mapping from UCS-4 form to UTF-16 form

UCS-4 (4-octet)	UTF-16, 2-octet elements
x = 0000 0000 .. 0000 FFFF (see Note 1)	x % 0001 0000;
x = 0001 0000 .. 0010 FFFF	y; z;
where	y = ((x - 0001 0000) / 400) + D800 z = ((x - 0001 0000) % 400) + DC00
x = 0011 0000 .. 7FFF FFFF	(no mapping (is defined)

NOTE - Code positions from 0000 D800 to 0000 DFFF are reserved for the UTF-16 form and do not occur in UCS-4. The values 0000 FFFE and 0000 FFFF also do not occur (see clause 8). The mapping of these code positions in UTF-16 is undefined.

Example:

The UCS-4 sequence [0000 0048] [0000 0069]
[0001 0000] [0000 0021] [0000 0021]

represents “Hi<0001 0000>!!”.

It is mapped to UTF-16 as:

[0048] [0069] [D800] [DC00] [0021] [0021]

If interpreted as UCS-2 this sequence will be

“Hi<RC-element from high-half zone>
<RC-element from low-half zone>!!”

C.4 Mapping from UTF-16 form to UCS-4 form

UTF-16, 2-octet elements UCS-4 (4-octet)

x = 0000; ... D7FF; x
x = E000; ... FFFF; x

pair (x, y) such that

x = D800; ... DBFF; ((x - D800) * 400
y = DC00; ... DFFF; + (y - DC00))
+ 0001 0000

Example:

The UTF-16 sequence

[0048] [0069] [D800] [DC00] [0021] [0021]

is mapped to UCS-4 as

[0000 0048] [0000 0069] [0001 0000]
[0000 0021] [0000 0021]

and represents “Hi<0001 0000>!!”.

C.5 Identification of UTF-16

When the escape sequences from ISO/IEC 2022 are used, the identification of UTF-16 and an implementation level (see clause 14) shall be by a designation sequence chosen from the following list:

ESC 02/05 02/15 04/10
UTF-16 with implementation level 1

ESC 02/05 02/15 04/11
UTF-16 with implementation level 2

ESC 02/05 02/15 04/12
UTF-16 with implementation level 3

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 15.

When the escape sequences from ISO 2022 are used, the identification of a return, or transfer, from UTF-16 to the coding system of ISO 2022 shall be as specified in 16.5 for a return or transfer from UCS.

C.6 Unpaired RC-elements: Interpretation by receiving devices

According to C.1 an unpaired RC-element (4.33) is not in conformance with the requirements of UTF-16.

If a receiving device that has adopted the UTF-16 form receives an unpaired RC-element because of error conditions either:

- in an originating device, or
- in the interchange between an originating and the receiving device, or
- in the receiving device itself,

then it shall interpret that unpaired RC-element in the same way that it interprets a character that is outside the adopted subset that has been identified for the device (see 2.3c).

NOTE - Since a high-half RC-element followed by a low-half RC-element is a sequence that is in accordance with UTF-16, the only possible type of syntactically malformed sequence is one or more unpaired RC-element.

Example:

A receiving/originating device which only handles the Basic Latin repertoire, and uses boxes (shown here as ◊) to display characters outside that repertoire, would display:

“The Greek letter Σ is the capital form of letter σ.”

as:

“The Greek letter ◊ is the capital form of letter ◊.”

Accordingly a similar device that can also interpret a UTF-16 data stream should also display an unpaired RC-element as a box.

C.7 Receiving devices, advisory notes

When a receiving device interprets a CC-data-element that is in accordance with UTF-16 the following advisory notes apply.

1. UTF-16 is designed to be compatible with the UCS-2 two-octet BMP Form (13.1). The high-half and low-half zones are assigned to separate ranges of code positions, to which characters can never be assigned. Thus the function of every RC-element (two-octet unit) within a UTF-16 data stream is always immediately identifiable from its value, without regard to context.

For example, the valid UTF-16 sequence [0048] [0069] [D800] [DC00] [0021] [0021] may also be interpreted by a receiving device that has adopted only UCS-2 as the coded representation of

“Hi<unrecognized><unrecognized>!!”

This form of compatibility is possible because RC-elements from the S-zone are interpreted according to UTF-16 by receiving devices that have adopted UTF-16, and as unrecognized characters by receiving devices that have only adopted UCS-2. Conse-

quently an originating device may transmit UTF-16 data even if the receiving device can only interpret that data as UCS-2 characters.

2. Designers of devices may choose to use UTF-16 as an internal representation for processing or other purposes. There are two primary issues for such devices:

- Does the device interpret (i.e., process according to the assigned semantics) some subset of the pairs (high-half + low-half) of RC-elements, e.g., render the pair as the intended single character?
- Does the device guarantee the integrity of every pair (high-half + low-half) of RC-elements, e.g., never separate such pairs in operations such as string truncation, insertion, or other modifications of the coded character sequence?

The decisions on these issues give rise to four possible combinations of capability in a device:

(U) UCS-2 implementations:

- Interpret no pairs.
- Do not guarantee integrity of pairs.

(W) Weak UTF-16 implementations:

- Interpret a non-null subset of pairs.
- Do not guarantee integrity of pairs.

(A) Aware UTF-16 implementations:

- Interpret no pairs.
- Guarantee integrity of pairs.

(S) Strong UTF-16 implementations:

- Interpret a non-null subset of pairs.
- Guarantee integrity of pairs.

Example:

The following sentence could be displayed in four different ways, assuming that both the weak and strong implementations have Etruscan fonts but no hieroglyphic fonts:

“The Greek letter Σ corresponds to <hieroglyphic-High> <hieroglyphic-Low> and to <Etruscan-High> <Etruscan-Low>.”

where <xxx-High> and <xxx-Low> represent RC-elements, from the High-half and Low-half zones respectively, corresponding to a character from the block indicated by xxx. These four ways are shown below.

U: “The Greek letter Σ corresponds to $\diamond\diamond$ and to $\diamond\diamond$.”

W: “The Greek letter Σ corresponds to $\diamond\diamond$ and to $\underline{\Sigma}$.”

A: “The Greek letter Σ corresponds to \diamond and to \diamond .”

S: “The Greek letter Σ corresponds to \diamond and to $\underline{\Sigma}$.”

where $\underline{\Sigma}$ here indicates the letter ES in the Etruscan font.

Annex D (normative)

UCS Transformation Format 8 (UTF-8)

UTF-8 is an alternative coded representation form for all of the characters of the UCS. It can be used to transmit text data through communication systems which assume that individual octets in the range 00 to 7F have a definition according to ISO/IEC 4873, including a C0 set of control functions according to the 8-bit structure of ISO/IEC 2022. UTF-8 also avoids the use of octet values in this range which have special significance during the parsing of file-name character strings in widely-used file-handling systems.

The number of octets in the UTF-8 coded representation of the characters of the UCS ranges from one to six; the value of the first octet indicates the number of octets in that coded representation.

D.1 Features of UTF-8

- UCS characters from the BASIC LATIN collection are represented in UTF-8 in accordance with ISO/IEC 4873, i.e. single octets with values ranging from 20 to 7E.
- Control functions in positions 0000 0000 to 0000 001F, and the DELETE character in position 0000 007F, are represented without the padding octets specified in clause 15, i.e. as single octets with values ranging from 00 to 1F, and 7F respectively in accordance with ISO/IEC 4873 and with the 8-bit structure of ISO/IEC 2022.
- Octet values 00 to 7F do not otherwise occur in the UTF-8 coded representation of any character. This provides compatibility with existing file-handling systems and communications sub-systems which parse CC-data-elements for these octet values.
- The first octet in the UTF-8 coded representation of any character can be directly identified when a CC-data-element is examined, one octet at a time, starting from an arbitrary location. It indicates the number of continuing octets (if any) in the multi-octet sequence that constitutes the coded representation of that character.

D.2 Specification of UTF-8

In the UTF-8 coded representation form each character from this International Standard shall have a coded representation that comprises a sequence of octets of length 1, 2, 3, 4, 5, or 6 octets.

For all sequences of one octet the most significant bit shall be a ZERO bit.

For all sequences of more than one octet, the number of consecutive ONE bits in the first octet, starting from the most significant bit position, shall indicate the number of octets in the sequence. The next most significant bit shall be a ZERO bit.

NOTE 1 - For example, the first octet of a 2-octet sequence has bits 110 in the most significant positions, and the first octet of a 6-octet sequence has bits 1111110 in the most significant positions.

All of the octets, other than the first in a sequence, are known as continuing octets. The two most significant bits of a continuing octet shall be a ONE bit followed by a ZERO bit.

The remaining bit positions in the octets of the sequence shall be "free bit positions" that are used to distinguish between the characters of this International Standard. These free bit positions shall be used, in order of increasing significance, for the bits of the UCS-4 coded representation of the character, starting from its least significant bit. Some of the high-order ZERO bits of the UCS-4 representation shall be omitted, as specified below.

Table D.1 below shows the format of the octets of a coded character according to UTF-8. Each free bit position available for distinguishing between the characters is indicated by an x. Each entry in the column "Maximum UCS-4 value" indicates the upper end of the range of coded representations from UCS-4 that may be represented in a UTF-8 sequence having the length indicated in the "Octet usage" column.

Table D.1 - Format of octets in a UTF-8 sequence

Octet usage	Format (binary)	No. of free bits	Maximum UCS-4 value
1 st of 1	0xxxxxxx	7	0000 007F
1 st of 2	110xxxxx	5	0000 07FF
1 st of 3	1110xxxx	4	0000 FFFF
1 st of 4	11110xxx	3	001F FFFF
1 st of 5	111110xx	2	03FF FFFF
1 st of 6	1111110x	1	7FFF FFFF
continuing) 2 nd .. 6 th)	10xxxxxx	6	

Table D.1 shows that, in a CC-data-element conforming to UTF-8, the range of values for each octet indicates its usage as follows:

- 00 to 7F first and only octet of a sequence;
- 80 to BF continuing octet of a multi-octet sequence;
- C0 to FD first octet of a multi-octet sequence;
- FE or FF not used.

The mapping between UCS-4 and UTF-8 shall be as shown in D.4; the reverse mapping is shown in D.5.

NOTE 2 - Examples of UCS-4 coded representations and the corresponding UTF-8 coded representations are shown in Tables D.2 and D.3.

Table D.2 shows the UCS-4 and the UTF-8 coded representations, in binary notation, for a selection of code positions from the UCS.

Table D.3 shows the UCS-4 and the UTF-8 coded representations, in hexadecimal notation, for the same selection of code positions from the UCS.

NOTE 3 - Control functions in positions 0000 0080 to 0000 009F are represented by two-octet sequences obtained by applying the rules specified in this clause to the four-octet padded forms of the control functions, i.e. such a control function is represented by a sequence in the range C2 80 to C2 9F.

**Table D.3 -
Examples in hexadecimal notation**

UCS-4 form	UTF-8 form
0000 0001;	01;
0000 007F;	7F;
0000 0080;	C2; 80;
0000 07FF;	DF; BF;
0000 0800;	E0; A0; 80;
0000 FFFF;	EF; BF; BF;
0001 0000;	F0; 90; 80; 80;
0010 FFFF;	F4; 8F; BF; BF;
001F FFFF;	F7; BF; BF; BF;
0020 0000;	F8; 88; 80; 80; 80;
03FF FFFF;	FB; BF; BF; BF; BF;
0400 0000;	FC; 84; 80; 80; 80; 80;
7FFF FFFF;	FD; BF; BF; BF; BF; BF;

Table D.2 - Examples in binary notation

Four-octet form - UCS-4	UTF-8 form
00000000 00000000 00000000 00000001;	00000001;
00000000 00000000 00000000 01111111;	01111111;
00000000 00000000 00000000 10000000;	11000010; 10000000;
00000000 00000000 00000111 11111111;	11011111; 10111111;
00000000 00000000 00001000 00000000;	11100000; 10100000; 10000000;
00000000 00000000 11111111 11111111;	11101111; 10111111; 10111111;
00000000 00000001 00000000 00000000;	11110000; 10010000; 10000000; 10000000;
00000000 00011111 11111111 11111111;	11110111; 10111111; 10111111; 10111111;
00000000 00100000 00000000 00000000;	11111000; 10001000; 10000000; 10000000; 10000000;
00000011 11111111 11111111 11111111;	11111011; 10111111; 10111111; 10111111; 10111111;
00000100 00000000 00000000 00000000;	11111100; 10000100; 10000000; 10000000; 10000000;
01111111 11111111 11111111 11111111;	11111101; 10111111; 10111111; 10111111; 10111111; 10111111;

D.3 Notation

1. All numbers are in hexadecimal notation, except for the decimal numbers used in the power-of operation (see 5 below).
2. Boundaries of code elements are indicated with semicolons; these are single-octet boundaries within UTF-8 coded representations, and four-octet boundaries within UCS-4 coded representations.
3. The symbol "%" indicates the modulo operation, e.g.: $7 \% 3 = 1$
4. The symbol "/" indicates the integer division operation, e.g.: $7 / 3 = 2$
5. Superscripting indicates the power-of operation, e.g.: $2^3 = 8$
6. Precedence is: power-of operation > integer division > modulo operation > integer multiplication > integer addition.

e.g.: $x / y^z \% w = ((x / (y^z)) \% w)$

D.4 Mapping from UCS-4 form to UTF-8 form

Table D.4 defines in mathematical notation the mapping from the UCS-4 coded representation form to the UTF-8 coded representation form.

In the left column (UCS-4) the notation x indicates the four-octet coded representation of a single character of the UCS. In the right column (UTF-8) x indicates the corresponding integer value.

NOTE 1 - Values of x in the range 0000 D800 .. 0000 DFFF are reserved for the UTF-16 form and do not occur in UCS-4. The mappings of these code positions in UTF-8 are undefined.

NOTE 2 - The algorithm for converting from UCS-4 to UTF-8 can be summarised as follows.

For each coded character in UCS-4 the length of octet sequence in UTF-8 is determined by the entry in the right column of Table D.1. The bits in the UCS-4 coded representation, starting from the least significant bit, are then distributed across the free bit positions in order of increasing significance until no more free bit positions are available.

D.5 Mapping from UTF-8 form to UCS-4 form

Table D.5 defines in mathematical notation the mapping from the UTF-8 coded representation form to the UCS-4 coded representation form.

In the left column (UTF-8) the following notations apply:

z is the first octet of a sequence. Its value determines the number of continuing octets in the sequence.

y is the 2nd octet in the sequence.

x is the 3rd octet in the sequence.

w is the 4th octet in the sequence.

v is the 5th octet in the sequence.

u is the 6th octet in the sequence.

The ranges of values applicable to these octets are shown in D.2 above, following Table D.1.

NOTE - The algorithm for converting from UTF-8 to UCS-4 can be summarised as follows.

For each coded character in UTF-8 the bits in the free bit positions are concatenated as a bit-string. The bits from this string, in increasing order of significance, are then distributed across the bit positions of a four-octet sequence, starting from the least significant bit position. The remaining bit positions of that sequence are filled with ZERO bits.

Range of values in UCS-4	Sequence of octets in UTF-8
x = 0000 0000 .. 0000 007F;	x;
x = 0000 0080 .. 0000 07FF;	$C0 + x / 2^6;$ $80 + x \% 2^6;$
x = 0000 0800 .. 0000 FFFF; (see Note 3)	$E0 + x / 2^{12};$ $80 + x / 2^6 \% 2^6;$ $80 + x \% 2^6;$
x = 0001 0000 .. 001F FFFF;	$F0 + x / 2^{18};$ $80 + x / 2^{12} \% 2^6;$ $80 + x / 2^6 \% 2^6;$ $80 + x \% 2^6;$
x = 0020 0000 .. 03FF FFFF;	$F8 + x / 2^{24};$ $80 + x / 2^{18} \% 2^6;$ $80 + x / 2^{12} \% 2^6;$ $80 + x / 2^6 \% 2^6;$ $80 + x \% 2^6;$
x = 0400 0000 .. 7FFF FFFF;	$FC + x / 2^{30};$ $80 + x / 2^{24} \% 2^6;$ $80 + x / 2^{18} \% 2^6;$ $80 + x / 2^{12} \% 2^6;$ $80 + x / 2^6 \% 2^6;$ $80 + x \% 2^6;$

Table D.5 - Mapping from UTF-8 to UCS-4

Sequence of octets in UTF-8	Four-octet sequences in UCS-4
$z = 00 \dots 7F;$	$z;$
$z = C0 \dots DF; y;$	$(z-C0)*2^6 + (y-80);$
$z = E0 \dots EF; y; x;$	$(z-E0)*2^{12} + (y-80)*2^6 + (x-80);$
$z = F0 \dots F7; y; x; w;$	$(z-F0)*2^{18} + (y-80)*2^{12} + (x-80)*2^6 + (w-80);$
$z = F8 \dots FB; y; x; w; v;$	$(z-F8)*2^{24} + (y-80)*2^{18} + (x-80)*2^{12} + (w-80)*2^6 + (v-80);$
$z = FC, FD; y; x; w; v; u;$	$(z-FC)*2^{30} + (y-80)*2^{24} + (x-80)*2^{18} + (w-80)*2^{12} + (v-80)*2^6 + (u-80);$

D.6 Identification of UTF-8

When the escape sequences from ISO/IEC 2022 are used, the identification of UTF-8 and an implementation level (see clause 14) shall be by a designation sequence chosen from the following list:

ESC 02/05 02/15 04/07
UTF-8 with implementation level 1

ESC 02/05 02/15 04/08
UTF-8 with implementation level 2

ESC 02/05 02/15 04/09
UTF-8 with implementation level 3

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 15.

When the escape sequences from ISO/IEC 2022 are used, the identification of a return, or transfer, from UTF-8 to the coding system of ISO/IEC 2022 shall be as specified in 16.5 for a return or transfer from UCS.

NOTE - The following escape sequence may also be used:

ESC 02/05 04/07 UTF-8.

The implementation level is not defined. The escape sequence used for a return to the coding system of ISO/IEC 2022 is not padded as specified in 16.5.

D.7 Incorrect sequences of octets: Interpretation by receiving devices

According to D.2 an octet in the range 00 to 7F or C0 to FB is the first octet of a UTF-8 sequence, and is followed by the appropriate number (from 0 to 5) of continuing octets in the range 80 to BF. Furthermore, octets whose value is FE or FF are not used; thus they are invalid in UTF-8.

If a CC-data-element includes either:

- a first octet that is not immediately followed by the correct number of continuing octets, or
- one or more continuing octets that are not required to complete a sequence of first and continuing octets, or
- an invalid octet,

then according to D.2 such a sequence of octets is not in conformance with the requirements of UTF-8. It is known as a malformed sequence.

If a receiving device that has adopted the UTF-8 form receives a malformed sequence, because of error conditions either:

- in an originating device, or
- in the interchange between an originating and a receiving device, or
- in the receiving device itself,

then it shall interpret that malformed sequence in the same way that it interprets a character that is outside the adopted subset that has been identified for the device (see 2.3c).

Annex E (informative)

Mirrored characters in bidirectional context

In the context of right-to-left (bidirectional) text, the following characters have semantic meaning. To preserve the meaning in right-to-left text, the graphic symbol representing the character may be rendered as the mirror image of the associated graphical symbol from the left-to-right context. These characters include mathematical symbols and paired characters such as the SQUARE BRACKETS. For example, in a right-to-left text segment, the GREATER-THAN SIGN (rendered as ">" in left-to-right text) may be rendered as the "<" graphic symbol.

0028	LEFT PARENTHESIS	2224	DOES NOT DIVIDE
0029	RIGHT PARENTHESIS	2226	NOT PARALLEL TO
003C	LESS-THAN SIGN	222B	INTEGRAL
003E	GREATER-THAN SIGN	222C	DOUBLE INTEGRAL
005B	LEFT SQUARE BRACKET	222D	TRIPLE INTEGRAL
005D	RIGHT SQUARE BRACKET	222E	CONTOUR INTEGRAL
007B	LEFT CURLY BRACKET	222F	SURFACE INTEGRAL
007D	RIGHT CURLY BRACKET	2230	VOLUME INTEGRAL
00AB	LEFT-POINTING DOUBLE ANGLE QUOTATION MARK	2231	CLOCKWISE INTEGRAL
00BB	RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK	2232	CLOCKWISE CONTOUR INTEGRAL
2039	SINGLE LEFT-POINTING ANGLE QUOTATION MARK	2233	ANTICLOCKWISE CONTOUR INTEGRAL
203A	SINGLE RIGHT-POINTING ANGLE QUOTATION MARK	2239	EXCESS
2045	LEFT SQUARE BRACKET WITH QUILL	223B	HOMOTHETIC
2046	RIGHT SQUARE BRACKET WITH QUILL	223C	TILDE OPERATOR
207D	SUPERSCRIPIT LEFT PARENTHESIS	223D	REVERSED TILDE
207E	SUPERSCRIPIT RIGHT PARENTHESIS	223E	INVERTED LAZY S
208D	SUBSCRIPT LEFT PARENTHESIS	223F	SINE WAVE
208E	SUBSCRIPT RIGHT PARENTHESIS	2240	WREATH PRODUCT
2201	COMPLEMENT	2241	NOT TILDE
2202	PARTIAL DIFFERENTIAL	2242	MINUS TILDE
2203	THERE EXISTS	2243	ASYMPTOTICALLY EQUAL TO
2204	THERE DOES NOT EXIST	2244	NOT ASYMPTOTICALLY EQUAL TO
2208	ELEMENT OF	2245	APPROXIMATELY EQUAL TO
2209	NOT AN ELEMENT OF	2246	APPROXIMATELY BUT NOT ACTUALLY EQUAL TO
220A	SMALL ELEMENT OF	2247	NEITHER APPROXIMATELY NOR ACTUALLY EQUAL TO
220B	CONTAINS AS MEMBER	2248	ALMOST EQUAL TO
220C	DOES NOT CONTAIN AS MEMBER	2249	NOT ALMOST EQUAL TO
220D	SMALL CONTAINS AS MEMBER	224A	ALMOST EQUAL OR EQUAL TO
2211	N-ARY SUMMATION	224B	TRIPLE TILDE
2215	DIVISION SLASH	224C	ALL EQUAL TO
2216	SET MINUS	2252	APPROXIMATELY EQUAL TO OR THE IMAGE OF
221A	SQUARE ROOT	2253	IMAGE OF OR APPROXIMATELY EQUAL TO
221B	CUBE ROOT	2254	COLON EQUALS
221C	FOURTH ROOT	2255	EQUALS COLON
221D	PROPORTIONAL TO	225F	QUESTIONED EQUAL TO
221F	RIGHT ANGLE	2260	NOT EQUAL TO
2220	ANGLE	2262	NOT IDENTICAL TO
2221	MEASURED ANGLE	2264	LESS-THAN OR EQUAL TO
2222	SPHERICAL ANGLE	2265	GREATER-THAN OR EQUAL TO
		2266	LESS-THAN OVER EQUAL TO
		2267	GREATER-THAN OVER EQUAL TO
		2268	LESS-THAN BUT NOT EQUAL TO
		2269	GREATER-THAN BUT NOT EQUAL TO
		226A	MUCH LESS-THAN
		226B	MUCH GREATER-THAN
		226E	NOT LESS-THAN
		226F	NOT GREATER-THAN
		2270	NEITHER LESS-THAN NOR EQUAL TO
		2271	NEITHER GREATER-THAN NOR EQUAL TO
		2272	LESS-THAN OR EQUIVALENT TO
		2273	GREATER-THAN OR EQUIVALENT TO

2274	NEITHER LESS-THAN NOR EQUIVALENT TO	22D8	VERY MUCH LESS-THAN
2275	NEITHER GREATER-THAN NOR EQUIVALENT TO	22D9	VERY MUCH GREATER-THAN
2276	LESS-THAN OR GREATER-THAN	22DA	LESS-THAN EQUAL TO OR GREATER-THAN
2277	GREATER-THAN OR LESS-THAN	22DB	GREATER-THAN EQUAL TO OR LESS-THAN
2278	NEITHER LESS-THAN NOR GREATER-THAN	22DC	EQUAL TO OR LESS-THAN
2279	NEITHER GREATER-THAN NOR LESS-THAN	22DD	EQUAL TO OR GREATER-THAN
227A	PRECEDES	22DE	EQUAL TO OR PRECEDES
227B	SUCCEEDS	22DF	EQUAL TO OR SUCCEEDS
227C	PRECEDES OR EQUAL TO	22E0	DOES NOT PRECEDE OR EQUAL
227D	SUCCEEDS OR EQUAL TO	22E1	DOES NOT SUCCEED OR EQUAL
227E	PRECEDES OR EQUIVALENT TO	22E2	NOT SQUARE IMAGE OF OR EQUAL TO
227F	SUCCEEDS OR EQUIVALENT TO	22E3	NOT SQUARE ORIGINAL OF OR EQUAL TO
2280	DOES NOT PRECEDE	22E4	SQUARE IMAGE OF OR NOT EQUAL TO
2281	DOES NOT SUCCEED	22E5	SQUARE ORIGINAL OF OR NOT EQUAL TO
2282	SUBSET OF	22E6	LESS-THAN BUT NOT EQUIVALENT TO
2283	SUPERSET OF	22E7	GREATER-THAN BUT NOT EQUIVALENT TO
2284	NOT A SUBSET OF	22E8	PRECEDES BUT NOT EQUIVALENT TO
2285	NOT A SUPERSET OF	22E9	SUCCEEDS BUT NOT EQUIVALENT TO
2286	SUBSET OF OR EQUAL TO	22EA	NOT NORMAL SUBGROUP OF
2287	SUPERSET OF OR EQUAL TO	22EB	DOES NOT CONTAIN AS NORMAL SUBGROUP
2288	NEITHER A SUBSET OF NOR EQUAL TO	22EC	NOT NORMAL SUBGROUP OF OR EQUAL TO
2289	NEITHER A SUPERSET OF NOR EQUAL TO	22ED	DOES NOT CONTAIN AS NORMAL SUBGROUP OR
228A	SUBSET OF WITH NOT EQUAL TO		EQUAL
228B	SUPERSET OF WITH NOT EQUAL TO	22F0	UP RIGHT DIAGONAL ELLIPSIS
228C	MULTISET	22F1	DOWN RIGHT DIAGONAL ELLIPSIS
228F	SQUARE IMAGE OF	2308	LEFT CEILING
2290	SQUARE ORIGINAL OF	2309	RIGHT CEILING
2291	SQUARE IMAGE OF OR EQUAL TO	230A	LEFT FLOOR
2292	SQUARE ORIGINAL OF OR EQUAL TO	230B	RIGHT FLOOR
2298	CIRCLED DIVISION SLASH	2320	TOP HALF INTEGRAL
22A2	RIGHT TACK	2321	BOTTOM HALF INTEGRAL
22A3	LEFT TACK	2329	LEFT-POINTING ANGLE BRACKET
22A6	ASSERTION	232A	RIGHT-POINTING ANGLE BRACKET
22A7	MODELS	3008	LEFT ANGLE BRACKET
22A8	TRUE	3009	RIGHT ANGLE BRACKET
22A9	FORCES	300A	LEFT DOUBLE ANGLE BRACKET
22AA	TRIPLE VERTICAL BAR TURNSTILE	300B	RIGHT DOUBLE ANGLE BRACKET
22AB	DOUBLE VERTICAL BAR DOUBLE RIGHT	300C	LEFT CORNER BRACKET
	TURNSTILE	300D	RIGHT CORNER BRACKET
22AC	DOES NOT PROVE	300E	LEFT WHITE CORNER BRACKET
22AD	NOT TRUE	300F	RIGHT WHITE CORNER BRACKET
22AE	DOES NOT FORCE	3010	LEFT BLACK LENTICULAR BRACKET
22AF	NEGATED DOUBLE VERTICAL BAR DOUBLE RIGHT	3011	RIGHT BLACK LENTICULAR BRACKET
	TURNSTILE	3014	LEFT TORTOISE SHELL BRACKET
22B0	PRECEDES UNDER RELATION	3015	RIGHT TORTOISE SHELL BRACKET
22B1	SUCCEEDS UNDER RELATION	3016	LEFT WHITE LENTICULAR BRACKET
22B2	NORMAL SUBGROUP OF	3017	RIGHT WHITE LENTICULAR BRACKET
22B3	CONTAINS AS NORMAL SUBGROUP	3018	LEFT WHITE TORTOISE SHELL BRACKET
22B4	NORMAL SUBGROUP OF OR EQUAL TO	3019	RIGHT WHITE TORTOISE SHELL BRACKET
22B5	CONTAINS AS NORMAL SUBGROUP OR EQUAL TO	301A	LEFT WHITE SQUARE BRACKET
22B6	ORIGINAL OF	301B	RIGHT WHITE SQUARE BRACKET
22B7	IMAGE OF	10300	OLD ITALIC LETTER A
22B8	MULTIMAP	10301	OLD ITALIC LETTER BE
22BE	RIGHT ANGLE WITH ARC	10302	OLD ITALIC LETTER KE
22BF	RIGHT TRIANGLE	10303	OLD ITALIC LETTER DE
22C9	LEFT NORMAL FACTOR SEMIDIRECT PRODUCT	10304	OLD ITALIC LETTER E
22CA	RIGHT NORMAL FACTOR SEMIDIRECT PRODUCT	10305	OLD ITALIC LETTER VE
22CB	LEFT SEMIDIRECT PRODUCT	10306	OLD ITALIC LETTER ZE
22CC	RIGHT SEMIDIRECT PRODUCT	10307	OLD ITALIC LETTER HE
22CD	REVERSE TILDE EQUALS	10308	OLD ITALIC LETTER THE
22D0	DOUBLE SUBSET	10309	OLD ITALIC LETTER I
22D1	DOUBLE SUPERSET	1030A	OLD ITALIC LETTER KA
22D6	LESS-THAN WITH DOT	1030B	OLD ITALIC LETTER EL
22D7	GREATER-THAN WITH DOT	1030C	OLD ITALIC LETTER EM

1030D OLD ITALIC LETTER EN
1030E OLD ITALIC LETTER ESH
1030F OLD ITALIC LETTER O
10310 OLD ITALIC LETTER PE
10311 OLD ITALIC LETTER SHE
10312 OLD ITALIC LETTER KU
10313 OLD ITALIC LETTER ER
10314 OLD ITALIC LETTER ES
10315 OLD ITALIC LETTER TE
10316 OLD ITALIC LETTER U
10317 OLD ITALIC LETTER EKS
10318 OLD ITALIC LETTER PHE

10319 OLD ITALIC LETTER KHE
1031A OLD ITALIC LETTER EF
1031B OLD ITALIC LETTER ERS
1031C OLD ITALIC LETTER CHE
1031D OLD ITALIC LETTER II
1031E OLD ITALIC LETTER UU
10320 OLD ITALIC NUMERAL ONE
10321 OLD ITALIC NUMERAL FIVE
10322 OLD ITALIC NUMERAL TEN
10323 OLD ITALIC FIFTY

Annex F (informative)

Alternate format characters

There is a special class of characters called Alternate Format Characters which are included for compatibility with some industry practices. These characters do not have printable graphic symbols, and are thus represented in the character code tables by dotted boxes.

The function of most of these characters is to indicate the correct presentation of a sequence of characters. For any text processing other than presentation (such as sorting and searching), the alternate format characters, except for ZWJ and ZWNJ described in F.1.1, can be ignored by filtering them out. The alternate format characters are not intended to be used in conjunction with bidirectional control functions from ISO/IEC 6429.

There are collections of graphic characters for selected subsets which consist of Alternate Format Characters (see annex A).

F.1 General format characters

F.1.1 Zero-width boundary indicators

COMBINING GRAPHEME JOINER (034F): The Combining Grapheme Joiner is used to indicate that adjacent characters belong to the same grapheme cluster. Grapheme clusters are sequences of one or more coded characters that correspond to what users think of as characters. They include, but are not limited to, composite sequences such as (g + °), digraphs such as Slovak “ch”, or sequences with letter modifiers such as k^w. The Combining Grapheme Joiner has no width in its presentation.

The following characters are used to indicate whether or not the adjacent characters are separated by a word boundary or hyphenation boundary. Each of these zero-width boundary indicators has no width in its usual own presentation.

SOFT HYPHEN (00AD): SOFT HYPHEN (SHY) is a graphic character, the visual representation of which is identical to that of HYPHEN, for use when an allowable automatic hyphenation line-break after it is to be indicated. Unless the SOFT HYPHEN occurs at the very end of a rendered line, the SOFT HYPHEN normally has zero width and no visible representation, and may also suppress the rendering of the following character.

NOTE – For example, for Swedish, “biljett<SHY>tång should be rendered as “biljettång” when there is no line-break after the SHY.

ARMENIAN HYPHEN (058A): This character has the same rule as the SOFT HYPHEN concerning its rendering but with a visual representation different from the HYPHEN.

MONGOLIAN TODO SOFT HYPHEN (1806): This character has the same rule as the SOFT HYPHEN concerning its rendering but with a visual representation different from the HYPHEN.

ZERO WIDTH SPACE (200B): This character behaves like a SPACE in that it indicates a word boundary, but unlike SPACE it has no presentational width. For example, this character could be used to indicate word boundaries in Thai, which does not use visible gaps to separate words.

ZERO WIDTH NO-BREAK SPACE (FEFF): This character behaves like a NO-BREAK SPACE in that it indicates the absence of word boundaries, but unlike NO-BREAK SPACE it has no presentational width. For example, this character could be inserted after the fourth character in the text “base+delta” to indicate that there is to be no word break between the “e” and the “+”.

NOTE - For additional usages of this character for “signature”, see annex H.

The following characters are used to indicate whether or not the adjacent characters are joined together in rendering (cursive joiners).

ZERO WIDTH NON-JOINER (200C): This character indicates that the adjacent characters are not joined together in cursive connection even when they would normally join together as cursive letter forms. For example, ZERO WIDTH NON-JOINER between ARABIC LETTER NOON and ARABIC LETTER MEEM indicates that the characters are not rendered with the normal cursive connection.

ZERO WIDTH JOINER (200D): This character indicates that the adjacent characters are represented with joining forms in cursive connection even when they would not normally join together as cursive letter forms. For example, in the sequence SPACE followed by ARABIC LETTER BEH followed by SPACE, ZERO WIDTH JOINER can be inserted between the first two characters to display the final form of the ARABIC LETTER BEH.

F.1.2 Format separators

The following characters are used to indicate formatting boundaries between lines or paragraphs.

LINE SEPARATOR (2028): This character indicates where a new line starts; although the text continues to the next line, it does not start a new paragraph; e.g. no inter-paragraph indentation might be applied.

PARAGRAPH SEPARATOR (2029): This character indicates where a new paragraph starts; e.g. the text continues on the next line and inter-paragraph line spacing or paragraph indentation might be applied.

F.1.3 Bidirectional text formatting

The following characters are used in formatting bidirectional text. If the specification of a subset includes these characters, then texts containing right-to-left characters are to be rendered with an implicit bidirectional algorithm.

An implicit algorithm uses the directional character properties to determine the correct display order of characters on a horizontal line of text.

The following characters are format characters that act exactly like right-to-left or left-to-right characters in terms of affecting ordering (Bidirectional format marks). They have no visible graphic symbols, and they do not have any other semantic effect.

Their use can be more convenient than the explicit embeddings or overrides, since their scope is more local.

LEFT-TO-RIGHT MARK (200E): In bidirectional formatting, this character acts like a left-to-right character (such as LATIN SMALL LETTER A).

RIGHT-TO-LEFT MARK (200F): In bidirectional formatting, this character acts like a right-to-left character (such as ARABIC LETTER NOON).

The following format characters indicate that a piece of text is to be treated as embedded, and is to have a particular ordering attached to it (Bidirectional format embeddings). For example, an English quotation in the middle of an Arabic sentence can be marked as being an embedded left-to-right string. These format characters nest in blocks, with the embedding and override characters initiating (pushing) a block, and the pop character terminating (popping) a block.

The function of the embedding and override characters are very similar; the main difference is that the embedding characters specify the implicit direction of the text, while the override characters specify the explicit direction of the text. When text has an explicit direction, the normal directional character properties are ignored, and all of the text is assumed to have the ordering direction determined by the override character.

LEFT-TO-RIGHT EMBEDDING (202A): This character is used to indicate the start of a left-to-right implicit embedding.

RIGHT-TO-LEFT EMBEDDING (202B): This character is used to indicate the start of a right-to-left implicit embedding.

LEFT-TO-RIGHT OVERRIDE (202D): This character is used to indicate the start of a left-to-right explicit embedding.

RIGHT-TO-LEFT OVERRIDE (202E): This character is used to indicate the start of a right-to-left explicit embedding.

POP DIRECTIONAL FORMATTING (202C): This character is used to indicate the termination of an implicit or explicit directional embedding initiated by the above characters.

F.1.4 Other boundary indicators

NARROW NO-BREAK SPACE (202F): This character is a non-breaking space. It is similar to 00A0 NO-BREAK SPACE, except that it is rendered with a narrower width. When used with the Mongolian script this character is usually rendered at one-third of the width of a normal space, and it separates a suffix from the Mongolian wordstem. This allows for the normal rules of Mongolian character shaping to apply, while indicating that there is no word boundary at that position.

F.2 Script-specific format characters

F.2.1 Hangul fill characters

The following format characters have a special usage for Hangul characters.

HANGUL FILLER (3164): This character represents the fill value used with the standard spacing Jamos.

HALFWIDTH HANGUL FILLER (FFA0): As with the other halfwidth characters, this character is included for compatibility with certain systems that provide halfwidth forms of characters.

F.2.2 Symmetric swapping format characters

The following characters are used in conjunction with the class of left/right handed pairs of characters listed in clause 19. The following format characters indicate whether the interpretation of the term LEFT or RIGHT in the character names is OPENING or CLOSING respectively. The following characters do not nest.

The default state of interpretation may be set by a higher level protocol or standard, such as ISO/IEC 6429. In the absence of such a protocol, the default state is as established by ACTIVATE SYMMETRIC SWAPPING.

INHIBIT SYMMETRIC SWAPPING (206A): Between this character and the following ACTIVATE SYMMETRIC SWAPPING format character (if any), the stored characters listed in clause 19 are interpreted and rendered as LEFT and RIGHT, and the processing specified in that clause is not performed.

ACTIVATE SYMMETRIC SWAPPING (206B): Between this character and the following INHIBIT SYMMETRIC

SWAPPING format character (if any), the stored characters listed in clause 19 are interpreted and rendered as OPENING and CLOSING characters as specified in that clause.

F.2.3 Character shaping selectors

The following characters are used in conjunction with Arabic presentation forms. During the presentation process, certain characters may be joined together in cursive connection or ligatures. The following characters indicate that the character shape determination process used to achieve this presentation effect is either activated or inhibited. The following characters do not nest.

INHIBIT ARABIC FORM SHAPING (206C): Between this character and the following ACTIVATE ARABIC FORM SHAPING format character (if any), the character shaping determination process is inhibited. The stored Arabic presentation forms are presented without shape modification. This is the default state.

ACTIVATE ARABIC FORM SHAPING (206D): Between this character and the following INHIBIT ARABIC FORM SHAPING format character (if any), the stored Arabic presentation forms are presented with shape modification by means of the character shaping determination process.

NOTE - These characters have no effect on characters that are not presentation forms: in particular, Arabic nominal characters as from 0600 to 06FF are always subject to character shaping, and are unaffected by these formatting characters.

F.2.4 Numeric shape selectors

The following characters allow the selection of the shapes in which the digits from 0030 to 0039 are rendered. The following characters do not nest.

NATIONAL DIGIT SHAPES (206E): Between this character and the following NOMINAL DIGIT SHAPES format character (if any), digits from 0030 to 0039 are rendered with the appropriate national digit shapes as specified by means of appropriate agreements. For example, they could be displayed with shapes such as the ARABIC-INDIC digits from 0660 to 0669.

NOMINAL DIGIT SHAPES (206F): Between this character and the following NATIONAL DIGIT SHAPES format character (if any), the digits from 0030 to 0039 are rendered with the shapes as those shown in the code tables for those digits. This is the default state.

F.2.5 Mongolian vowel separator

MONGOLIAN VOWEL SEPARATOR (180E): This character may be used between the MONGOLIAN LETTER A or the MONGOLIAN LETTER E at the end of a word and the preceding consonant letter. It indicates a special form of the graphic symbol for the letter A or E and the preceding consonant. When rendered in visible form it is generally shown as a narrow space between the letters, but it may sometimes be shown as a distinct graphic symbol to assist the user.

F.3 Ideographic description characters

An Ideographic Description Character (IDC) is a graphic character, which is used with a sequence of other graphic characters to form an Ideographic Description Sequence (IDS). Such a sequence may be used to describe an ideographic character which is not specified within this International Standard.

The IDS describes the ideograph in the abstract form. It is not interpreted as a composed character and does not imply any specific form of rendering.

NOTE - An IDS is not a character and therefore is not a member of the repertoire of ISO/IEC 10646.

F.3.1 Syntax of an ideographic description sequence

An IDS consists of an IDC followed by a fixed number of Description Components (DC). A DC may be any one of the following :

- a coded ideograph
- a coded radical
- another IDS

NOTE - The above description implies that any IDS may be nested within another IDS.

Each IDC has four properties as summarized in table F.1 below;

- the number of DCs used in the IDS that commences with that IDC,
- the definition of its acronym,
- the syntax of the corresponding IDS,
- the relative positions of the DCs in the visual representation of the ideograph that is being described in its abstract form.

The syntax of the IDS introduced by each IDC is indicated in the "IDS Acronym and Syntax" column of the table by the abbreviated name of the IDC (e.g. IDC-LTR) followed by the corresponding number of DCs, i.e. (D₁ D₂) or (D₁ D₂ D₃).

NOTE - An IDS is restricted to no more than 16 characters in length. Also no more than six ideographs and/or radicals may occur between any two instances of an IDC character within an IDS.

F.3.2 Individual definitions of the ideographic description characters

IDEOGRAPHIC DESCRIPTION CHARACTER LEFT TO RIGHT (2FF0): The IDS introduced by this character describes the abstract form of the ideograph with D₁ on the left and D₂ on the right.

IDEOGRAPHIC DESCRIPTION CHARACTER ABOVE TO BELOW (2FF1): The IDS introduced by this character describes the abstract form of the ideograph with D₁ above D₂.

IDEOGRAPHIC DESCRIPTION CHARACTER LEFT TO MIDDLE AND RIGHT (2FF2): The IDS introduced by this character describes the abstract form of the ideograph with D₁ on the left of D₂, and D₂ on the left of D₃.

IDEOGRAPHIC DESCRIPTION CHARACTER ABOVE TO MIDDLE AND BELOW (2FF3): The IDS introduced by this character describes the abstract form of the ideograph with D_1 above D_2 , and D_2 above D_3 .

IDEOGRAPHIC DESCRIPTION CHARACTER FULL SURROUND (2FF4): The IDS introduced by this character describes the abstract form of the ideograph with D_1 surrounding D_2 .

IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM ABOVE (2FF5): The IDS introduced by this character describes the abstract form of the ideograph with D_1 above D_2 , and surrounding D_2 on both sides.

IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM BELOW (2FF6): The IDS introduced by this character describes the abstract form of the ideograph with D_1 below D_2 , and surrounding D_2 on both sides.

IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM LEFT (2FF7): The IDS introduced by this character describes the abstract form of the ideograph with D_1 on the left of D_2 , and surrounding D_2 above and below.

IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM UPPER LEFT (2FF8): The IDS introduced by this character describes the abstract form of the ideograph with D_1 at the top left corner of D_2 , and partly surrounding D_2 above and to the left.

IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM UPPER RIGHT (2FF9): The IDS introduced by this character describes the abstract form of the ideograph with D_1 at the top right corner of D_2 , and partly surrounding D_2 above and to the right.

IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM LOWER LEFT (2FFA): The IDS introduced by this character describes the abstract form of the ideograph with D_1 at the bottom left corner of D_2 , and partly surrounding D_2 below and to the left.

IDEOGRAPHIC DESCRIPTION CHARACTER OVERLAID (2FFB): The IDS introduced by this character describes the abstract form of the ideograph with D_1 and D_2 overlaying each other.

F.4 Interlinear annotation characters

The following characters are used to indicate that an identified character string (the annotation string) is regarded as providing an annotation for another identified character string (the base string).

INTERLINEAR ANNOTATION ANCHOR (FFF9): This character indicates the beginning of the base string.

INTERLINEAR ANNOTATION SEPARATOR (FFFA): This character indicates the end of the base string and the beginning of the annotation string.

INTERLINEAR ANNOTATION TERMINATOR (FFFB): This character indicates the end of the annotation string.

The relationship between the annotation string and the base string is defined by agreement between the user of the originating device and the user of the receiving device. For example, if the base string is rendered in a visible form the annotation string may be rendered on a different line from the base string, in a position close to the base string.

If the interlinear annotation characters are filtered out during processing, then all characters between the Interlinear Annotation Separator and the Interlinear Annotation Terminator should also be filtered out.

F.5 Subtending format characters

The following characters are used to subtend a sequence of subsequent characters:

0600	ARABIC NUMBER SIGN
0601	ARABIC SIGN SANAH
0602	ARABIC FOOTNOTE MARKER
06DD	ARABIC END OF AYAH
070F	SYRIAC ABBREVIATION MARK

The scope of these characters is the subsequent sequence of digits (plus certain other characters), with the exact specification as defined in the Unicode Standard, Version 3.2, for ARABIC END OF AYAH.

Annex G (informative)

Alphabetically sorted list of character names

The alphabetically sorted list of character names is provided in machine-readable format that is accessible as a link to this document. The content linked to is a plain text file, using ISO/IEC 646-IRV characters with LINE FEED as end of line mark, that specifies, after a 4-lines header, all the character names from ISO/IEC 10646 except Hangul syllables and CJK-ideographs (these are characters from blocks:

HANGUL SYLLABLES,
CJK UNIFIED IDEOGRAPHS,
CJK UNIFIED IDEOGRAPHS EXTENSION A,
CJK UNIFIED IDEOGRAPHS EXTENSION B,
CJK COMPATIBILITY IDEOGRAPHS and
CJK COMPATIBILITY IDEOGRAPHS SUPPLEMENT).

The format of the file, after the header, is as follows:

01-05 octet: UCS-4 five-digit abbreviated form,

06 octet: TAB character,

07-end of line: character name

[Click on this highlighted text to access the reference file.](#)

NOTE 1 – The content is also available as a separate viewable file in the same file directory as this document. The file is named: "Allnames.txt".

NOTE 2 – The referenced files are only available to users who obtain their copy of the standard in a machine-readable format. However, the file format makes them printable.

Annex H (informative)

The use of “signatures” to identify UCS

This annex describes a convention for the identification of features of the UCS, by the use of “signatures” within data streams of coded characters. The convention makes use of the character ZERO WIDTH NO-BREAK SPACE, and is applied by a certain class of applications.

When this convention is used, a signature at the beginning of a stream of coded characters indicates that the characters following are encoded in the UCS-2 or UCS-4 coded representation, and indicates the ordering of the octets within the coded representation of each character (see 6.3). It is typical of the class of applications mentioned above, that some make use of the signatures when receiving data, while others do not. The signatures are therefore designed in a way that makes it easy to ignore them.

In this convention, the ZERO WIDTH NO-BREAK SPACE character has the following significance when it is present at the beginning of a stream of coded characters:

UCS-2 signature: FEFF

UCS-4 signature: 0000 FEFF

UTF-8 signature: EF BB BF

UTF-16 signature: FEFF

An application receiving data may either use these signatures to identify the coded representation form, or may ignore them and treat FEFF as the ZERO WIDTH NO-BREAK SPACE character.

If an application which uses one of these signatures recognizes its coded representation in reverse sequence (e.g. hexadecimal FFFE), the application can identify that the coded representations of the following characters use the opposite octet sequence to the sequence expected, and may take the necessary action to recognize the characters correctly.

NOTE - The hexadecimal value FFFE does not correspond to any coded character within ISO/IEC 10646.

Annex J (informative)

Recommendation for combined receiving/originating devices with internal storage

This annex is applicable to a widely-used class of devices that can store received CC-data elements for subsequent retransmission.

This recommendation is intended to ensure that loss of information is minimized between the receipt of a CC-data-element and its retransmission.

A device of this class includes a receiving device component and an originating device component as in 2.3, and can also store received CC-data-elements for retransmission, with or without modification by the actions of the user on the corresponding characters represented within it. Within this class of device, two distinct types are identified here, as follows.

1. Receiving device with full retransmission capability
The originating device component will retransmit the coded representations of any received characters, including those that are outside the identified subset of the receiving device component, without change to their coded representation, unless modified by the user.
2. Receiving device with subset retransmission capability
The originating device component can re-transmit only the coded representations of the characters of the subset adopted by the receiving device component.

Annex K (informative)

Notations of octet value representations

Representation of octet values in ISO/IEC 10646 except in clause 16 is different from other character coding standards such as ISO/IEC 2022, ISO/IEC 6429 and ISO 8859. This annex clarifies the relationship between the two notations.

- In ISO/IEC 10646, the notation used to express an octet value is z , where z is a hexadecimal number in the range 00 to FF.

For example, the character ESCAPE (ESC) of ISO/IEC 2022 is represented by 1B.

- In other character coding standards, the notation used to express an octet value is x/y , where x and y are two decimal numbers in the range 00 to 15. The correspondence between the notations of the form x/y and the octet value is as follows.

x is the number represented by bit 8, bit 7, bit 6 and bit 5 where these bits are given the weight 8, 4, 2 and 1 respectively;

y is the number represented by bit 4, bit 3, bit 2 and bit 1 where these bits are given the weight 8, 4, 2 and 1 respectively.

For example, the character ESC of ISO/IEC 2022 is represented by 01/11.

Thus ISO/IEC 2022 (and other character coding standards) octet value notation can be converted to ISO/IEC 10646 octet value notation by converting the value of x and y to hexadecimal notation. For example; 04/15 is equivalent to 4F.

Annex L (informative)

Character naming guidelines

Guidelines for generating and presenting unique names of characters in ISO/IEC JTC1/SC2 standards are listed in this annex for information. These guidelines are used in information technology coded character set standards such as ISO/IEC 646, ISO/IEC 6937, ISO/IEC 8859, ISO/IEC 10367 as well as in ISO/IEC 10646.

These Guidelines specify rules for generating and presenting unique names of characters in those versions of the standards that are in the English language.

NOTE. In a version of such a standard in another language:

- a) these rules may be amended to permit names of characters to be generated using words and syntax that are considered appropriate within that language;
- b) the names of the characters from this version of the standard may be replaced by equivalent unique names constructed according to the rules amended as in a) above.

Rules 1 to 3 are implemented without exceptions. However it must be accepted that in some cases (e.g. historical or traditional usage, unforeseen special cases, and difficulties inherent to the nature of the character considered), exceptions to some of the other rules will have to be tolerated. Nonetheless, these rules are applied wherever possible.

Rule 1

By convention, only Latin capital letters A to Z, space, and hyphen are used for writing the names of characters.

NOTE - Names of characters may also include digits 0 to 9 (provided that a digit is not the first character in a word) if inclusion of the name of the corresponding digit(s) would be inappropriate. As an example the name of the character at position 201A is SINGLE LOW-9 QUOTATION MARK; the symbol for the digit 9 is included in this name to illustrate the shape of the character, and has no numerical significance.

Rule 2

The names of control functions are coupled with an acronym consisting of Latin capital letters A to Z and, where required, digits. Once the name has been specified for the first time, the acronym may be used in the remainder of the text where required for simplification and clarity of the text. Exceptionally, acronyms may be used for graphic characters where usage already exists and clarity requires it, in particular in code tables.

Examples:

Name: LOCKING-SHIFT TWO RIGHT

Acronym: LS2R

Name: SOFT HYPHEN

Acronym: SHY

NOTE - In ISO/IEC 6429, also the names of the modes have been presented in the same way as control functions.

Rule 3

In some cases, the name of a character can be followed by an additional explanatory statement not part of the name. These statements are in parentheses and not in capital Latin letters except the initials of the word where required. See examples in rule 12.

The name of a character may also be followed by a single * symbol not part of the name. This indicates that additional information on the character appears in Annex P. Any * symbols are omitted from the character names listed in Annex G.

Rule 4

The name of a character wherever possible denotes its customary meaning, for example PLUS SIGN. Where this is not possible, names describe shapes, not usage; for example: UPWARDS ARROW.

The name of a character is not intended to identify its properties or attributes, or to provide information on its linguistic characteristics, except as defined in Rule 6 below.

Rule 5

Only one name is given to each character.

Rule 6

The names are constructed from an appropriate set of the applicable terms of the following grid and ordered in the sequence of this grid. Exceptions are specified in Rule 11. The words WITH and AND may be included for additional clarity when needed.

1	Script	5	Attribute
2	Case	6	Designation
3	Type	7	Mark(s)
4	Language	8	Qualifier

Examples of such terms:

Script	Latin, Cyrillic, Arabic
Case	capital, small
Type	letter, ligature, digit
Language	Ukrainian
Attribute	final, sharp, subscript, vulgar
Designation	customary name, name of letter
Mark	acute, ogonek, ring above, diaeresis
Qualifier	sign, symbol

Examples of names:

LATIN CAPITAL LETTER A WITH ACUTE

1 2 3 6 7

DIGIT FIVE

3 6

LEFT CURLY BRACKET

5 5 6

NOTE 1 - A ligature is a graphic symbol in which two or more other graphic symbols are imaged as a single graphic symbol.

NOTE 2 - Where a character comprises a base letter with multiple marks, the sequence of those in the name is the order in which the marks are positioned relative to the base letter, starting with the marks above the letters taken in upwards sequence, and followed by the marks below the letters taken in downwards sequence.

Rule 7

The letters of the Latin script are represented within their name by their basic graphic symbols (A, B, C, etc.). The letters of all other scripts are represented by their transcription in the language of the first published International Standard.

Examples:

K LATIN CAPITAL LETTER K

Ю CYRILLIC CAPITAL LETTER YU

Rule 8

In principle when a character of a given script is used in more than one language, no language name is specified. Exceptions are tolerated where an ambiguity would otherwise result.

Examples:

И CYRILLIC CAPITAL LETTER I

I CYRILLIC CAPITAL LETTER
BYELORUSSIAN-UKRAINIAN I

Rule 9

Letters that are elements of more than one script are considered different even if their shape is the same; they have different names.

Examples:

À LATIN CAPITAL LETTER A
Α GREEK CAPITAL LETTER ALPHA
А CYRILLIC CAPITAL LETTER A

Rule 10

A character of one script used in isolation in another script, for example as a graphic symbol in relation with physical units of dimension, is considered as a character different from the character of its native script.

Example:

μ MICRO SIGN

Rule 11

A number of characters have a traditional name consisting of one or two words. It is not intended to change this usage.

Examples:

' APOSTROPHE
: COLON
@ COMMERCIAL AT
— LOW LINE
~ TILDE

Rule 12

In some cases, characters of a given script, often punctuation marks, are used in another script for a different usage. In these cases the customary name reflecting the most general use is given to the character. The customary name may be followed in the list of characters of a particular standard by the name in parentheses which this character has in the script specified by this particular standard.

Example:

⸱ UNDERTIE (Enotikon)

Rule 13

The above rules do not apply to ideographic characters. These characters are identified by alpha-numeric identifiers specified for each ideographic character (see clause 27).

Annex M (informative)

Sources of characters

Several sources and contributions were used for constructing this coded character set. In particular, characters of the following national and international standards are included in ISO/IEC 10646.

ISO 233:1984, *Documentation - Transliteration of Arabic characters into Latin characters*.

ISO/IEC 646:1991, *Information technology - ISO 7-bit coded character set for information interchange*.

ISO 2033:1983, *Information processing - Coding of machine readable characters (MICR and OCR)*.

ISO 2047:1975, *Information processing - Graphical representations for the control characters of the 7-bit coded character set*.

ISO 5426:1983, *Extension of the Latin alphabet coded character set for bibliographic information interchange*.

ISO 5427:1984, *Extension of the Cyrillic alphabet coded character set for bibliographic information interchange*.

ISO 5428:1984, *Greek alphabet coded character set for bibliographic information interchange*.

ISO 6438:1983, *Documentation - African coded character set for bibliographic information interchange*.

ISO 6861, *Information and documentation - Glagolitic coded character set for bibliographic information interchange*.

ISO 6862, *Information and documentation - Mathematical coded character set for bibliographic information interchange*.

ISO 6937:1994, *Information technology - Coded graphic character sets for text communication - Latin alphabet*.

ISO/IEC 8859, *Information technology - 8-bit single-byte coded graphic character sets*

-Part 1: *Latin alphabet No. 1 (1998)*.

-Part 2: *Latin alphabet No. 2 (1999)*.

-Part 3: *Latin alphabet No. 3 (1999)*.

-Part 4: *Latin alphabet No. 4 (1998)*.

-Part 5: *Latin/Cyrillic alphabet (1999)*

-Part 6: *Latin/Arabic alphabet (1999)*

-Part 7: *Latin/Greek alphabet*

-Part 8: *Latin/Hebrew alphabet (1999)*

-Part 9: *Latin alphabet No. 5 (1999)*

-Part 10: *Latin alphabet No. 6 (1998)*.

ISO 8879:1986, *Information processing - Text and office systems - Standard Generalized Markup Language (SGML)*.

ISO 8957:1996, *Information and documentation - Hebrew alphabet coded character sets for bibliographic information interchange*.

ISO 9036:1987, *Information processing - Arabic 7-bit coded character set for information interchange*.

ISO/IEC 9995-7:1994, *Information technology - Keyboard layouts for text and office systems - Part 7: Symbols used to represent functions*.

ISO/IEC 10367:1991, *Information technology - Standardized coded graphic character sets for use in 8-bit codes*.

ISO 10754:1984, *Information and documentation - Extension of the Cyrillic alphabet coded character set for non-Slavic languages for bibliographic information interchange*.

ISO 11548-1:2001, *Communication aids for blind persons - identifiers, names and assignation to coded character sets for 8-dot Braille characters - Part 1: General guidelines for Braille identifiers and shift marks*.

ISO/IEC TR 15285:1998, *Information technology - An operational model for characters and glyphs*.

ISO international register of character sets to be used with escape sequences. (registration procedure ISO 2375:1985) .

ANSI X3.4-1986 American National Standards Institute. *Coded character set - 7-bit American national standard code*.

ANSI X3.32-1973 American National Standards Institute. *American national standard graphic representation of the control characters of American national standard code for information interchange*.

ANSI Y10.20-1988 American National Standards Institute. *Mathematic signs and symbols for use in physical sciences and technology.*

ANSI Y14.5M-1982 American National Standard. *Engineering drawings and related document practices, dimensioning and tolerances.*

ANSI Z39.47-1985 American National Standards Institute. *Extended Latin alphabet coded character set for bibliographic use.*

ANSI Z39.64-1989 American National Standards Institute. *East Asian character code for bibliographic use.*

ASMO 449-1982 Arab Organization for Standardization and Metrology. *Data processing - 7-bit coded character set for information interchange.*

GB2312-80 *Code of Chinese Graphic Character Set for Information Interchange: Jishu Biaozhun Chubanshe* (Technical Standards Publishing).

NOTE - For additional sources of the CJK unified ideographs in ISO/IEC 10646 refer to clause 27.

GB13134: *Xinxi jiaohuanyong yiwen bianma zifuji (Yi coded character set for information interchange)*, [prepared by] Sichuansheng minzushiwu weiyuanhui. Beijing, Jishu Biaozhun Chubanshe (Technical Standards Press), 1991. (GB 13134-1991).

GBK (*Guo Biao Kuo*) *Han character internal code extension specification: Jishu Biaozhun Chubanshe* (Technical Standards Publishing, Beijing)

IS 13194:1991 Bureau of Indian Standards *Indian script code for information interchange - ISCII*

LTD 37(1610)-1988 *Indian standard code for information interchange.*

I. S. 434:1999, *Information Technology - 8-bit single-byte graphic coded character set for Ogham = Teicneolaíocht Eolais - Tacar carachtar grafach Oghaim códaíthe go haonbheartach le 8 ngiotán.* National Standards Authority of Ireland.

JIS X 0201-1976 Japanese Standards Association. *Jouhou koukan you fugou (Code for Information Interchange).*

JIS X 0208-1990 Japanese Standards Association. *Jouhou koukan you kanji fugoukei (Code of the Japanese Graphic Character Set for Information Interchange).*

JIS X 0212-1990 Japanese Standards Association. *Jouhou koukan you kanji fugou-hojo kanji (Code of the supplementary Japanese graphic character set for information interchange).*

JIS X 0213:2000, Japanese Standards Association. *7-bit and 8-bit double byte coded extended KANJI sets for information interchange, 2000-01-20.*

KS C 5601-1992 Korean Industrial Standards Association. *Jeongbo gyohwanyong buho (Code for Information Interchange).*

LVS 18-92 Latvian National Centre for Standardization and Metrology *Libiesu kodu tabula ar 191 simbolu.*

SI 1311.2 - 1996 The Standards Institution of Israel Information Technology. *ISO 8-bit coded character set for information interchange with Hebrew points and cantillation marks.*

SLS 1134:1996 Sri Lanka Standards Institution *Sinhala character code for information interchange.*

TIS 620-2533 *Thai Industrial Standard for Thai Character Code for Computer.* (1990)

The following publications were also used as sources of characters for the Basic Multilingual Plane.

Allworth, Edward. *Nationalities of the Soviet East: Publications and Writing Systems.* New York, London, Columbia University Press, 1971. ISBN 0-231-03274-9.

Armbruster, Carl Hubert. *Initia Amharica: an Introduction to Spoken Amharic.* Cambridge, Cambridge University Press, 1908-20.

Barry, Randall K. 1997. *ALA-LC romanization tables: transliteration schemes for non-Roman scripts.* Washington, DC: Library of Congress Cataloging Distribution Service. ISBN 0-8444-0940-5

Benneth, Solbritt, Jonas Ferenius, Helmer Gustavson, & Marit Åhlén. 1994. *Runmärkt: från brev till klotter. Runorna under medeltiden.* [Stockholm]: Carlsson Bokförlag. ISBN 91-7798-877-9

Beyer, Stephen V. *The classical Tibetan language.* State University of New York. ISBN 0-7914-1099-4

Bburx Ddie Su (= Bian Xiezhe). 1984. *Nuo-su bbur-ma shep jie zzit: Syp-chuo se nuo bbur-ma syt mu curx su niep sha zho ddop ma bbur-ma syt mu wo yuop hop, Bburx Ddie da Su.* [Chengdu]: Syp-chuo co cux tep yy ddurx dde. *Yi wen jian zi ben: Yi Han wen duizhao ban.* Chengdu: Sichuan minzu chubanshe. [An examination of the fundamentals of the Yi script. Chengdu: Sichuan National Press.]

Bburx Ddie Su. *Nip huo bbur-ma ssix jie: Nip huo bbur-ma ssi jie Bburx Ddie curx Su. = Yi Han zidian.* Chengdu: Sichuan minzu chubanshe, 1990. ISBN 7-5409-0128-4

Daniels, Peter T., and William Bright, eds. 1996. *The world's writing systems.* New York; Oxford: Oxford University Press. ISBN 0-19-507993-0

Derolez, René. 1954. *Runica manuscripta: the English tradition.* (Rijksuniversiteit te Gent: Werken uitgegeven door de Faculteit van de Wijsbegeerte en Letteren; 118e aflevering) Brugge: De Tempel.

Diringer, David. 1996. *The alphabet: a key to the history of mankind.* New Delhi: Munshiram Manoharlal. ISBN 81-215-0780-0

- Esling, John. *Computer coding of the IPA: supplementary report*. Journal of the International Phonetic Association, 20:1 (1990), p. 22-26.
- Faulmann, Carl. 1990 (1880). *Das Buch der Schrift*. Frankfurt am Main: Eichborn. ISBN 3-8218-1720-8
- Friesen, Otto von. *Runorna*. Stockholm, A. Bonnier [1933]. (Nordisk kultur, 6).
- Geiger, Wilhelm. *Maldivian Linguistic Studies*. New Delhi, Asian Educational Services, 1996. ISBN 81-206-1201-9.
- Gunasekara, Abraham Mendis. 1986 (1891). *A comprehensive grammar of the Sinhalese language*. New Delhi: Asian Educational Services.
- Haarmann, Harald. 1990. *Universalgeschichte der Schrift*. Frankfurt/Main; New York: Campus. ISBN 3-593-34346-0
- Holmes, Ruth Bradley, and Betty Sharp Smith. 1976. *Beginning Cherokee: Talisgo galiquogi dideliquasdodi Tsalagi digoweli*. Norman: University of Oklahoma Press.
- International Phonetic Association. The IPA 1989 Kiel Convention Workgroup 9 report: *Computer Coding of IPA Symbols and Computer Representation of Individual Languages*. Journal of the International Phon. Assoc., 19:2 (1989), p. 81-82.
- Imprimerie Nationale. 1990. *Les caractères de l'Imprimerie Nationale*. Paris: Imprimerie Nationale Éditions. ISBN 2-11-081085-8
- International Phonetic Association. *The International Phonetic Alphabet* (revised to 1989).
- Jensen, Hans. 1969. *Die Schrift in Vergangenheit und Gegenwart*. 3., neubearbeitete und erweiterte Auflage. Berlin: VEB Deutscher Verlag der Wissenschaften.
- Kefarnissy, Paul. *Grammaire de la langue araméenne syriaque*. Beyrouth, 1962.
- Knuth, Donald E. *The TeXbook*. – 19th. printing, rev. – Reading, MA : Addison-Wesley, 1990.
- Kuruch, Rimma Dmitrievna. *Saamsko-russkiy slovar'*. Moskva: Russkiy iazyk. 1985
- Launhardt, Johannes. *Guide to Learning the Oromo (Galla) Language*. Addis Ababa, Launhardt [1973?]
- Leslau, Wolf. *Amharic Textbook*. Weisbaden, Harrassowitz; Berkeley, University of California Press, 1968.
- Mandarin Promotion Council, Ministry of Education, Taiwan. *Shiangtu yuyan biauyin fuhau shoutse (The Handbook of Taiwan Languages Phonetic Alphabet)*. 1999.
- Nakanishi, Akira. 1990. *Writing systems of the world: alphabets, syllabaries, pictograms*. Rutland, VT: Charles E. Tuttle. ISBN 0-8048-1654-9
- Okell, John. 1971. *A guide to the romanization of Burmese*. (James G. Forlang Fund; 27) London: Royal Asiatic Society of Great Britain and Ireland.
- Page, R. I. 1987. *Runes*. (Reading the Past; 4) Berkeley & Los Angeles: University of California Press. ISBN 0-520-06114-4
- Pullum, Geoffrey K. *Phonetic symbol guide*. Geoffrey K. Pullum and William A. Ladusaw. – Chicago : University of Chicago Press, 1986.
- Pullum, Geoffrey K. *Remarks on the 1989 revision of the International Phonetic Alphabet*. Journal of the International Phonetic Association, 20:1 (1990), p. 33-40.
- Roop, D. Haigh. 1972. *An introduction to the Burmese writing system*. New Haven and London: Yale University Press. ISBN 0-300-01528-3
- Santos, Hector. 1994. *The Tagalog script*. (Ancient Philippine Scripts Series; 1). Los Angeles: Sushi Dog Graphics.
- Santos, Hector. 1995. *The living scripts*. (Ancient Philippine Scripts Series; 2). Los Angeles: Sushi Dog Graphics.
- Selby, Samuel M. *Standard mathematical tables*. – 16th ed. – Cleveland, OH : Chemical Rubber Co., 1968. Shepherd, Walter.
- Shepherd, Walter. *Shepherd's glossary of graphic signs and symbols*. Compiled and classified for ready reference. – New York : Dover Publications, [1971].
- Shinmura, Izuru. *Kojien – Dai 4-han*. – Tokyo : Iwanami Shoten, Heisei 3 [1991].
- The Unicode Consortium *The Unicode Standard. Worldwide Character Encoding Version 1.0, Volume One*. – Reading, MA : Addison-Wesley, 1991.
- The Unicode Consortium *The Unicode standard, Version 2.0*. Reading, MA: Addison-Wesley, 1996. ISBN 0-201-48345-9
- The Unicode Consortium *The Unicode standard, Version 3.0*. Reading, MA: Addison-Wesley Developer's Press, 2000. ISBN 0-201-61633-5
- The Unicode Consortium *The Unicode standard, Version 4.0*. Reading, MA: Addison-Wesley Developer's Press, 2000. ISBN (TBD) FORTHCOMING

The following publications were also used as sources of characters for the Supplementary Multilingual Plane.

Deseret

Ivins, Stanley S. "The Deseret Alphabet" *Utah Humanities Review* 1 (1947):223-39.

Old Italic

Bonfante, Larissa. 1996. "The scripts of Italy", in Peter T. Daniels and William Bright, eds. *The world's writing sys-*

tems. New York; Oxford: Oxford University Press. ISBN 0-19-507993-0

Gothic

Fairbanks, Sydney, and F. P. Magoun Jr. 1940. 'On writing and printing Gothic', in *Speculum* 15:313-16.

Byzantine Musical Symbols

ELOT 1373. *The Greek Byzantine Musical Notation System*. Athens, 1997 (ΣΕΠ ΕΛΟΤ 1373: 1997).

Musical Symbols

Heussenstamm, George. *Norton Manual of Music Notation*. New York: W. W. Norton, 1987

Rastall, Richard. *Notation of Western Music: An Introduction*. London: Dent, 1983.

Annex N (informative)

External references to character repertoires

N.1 Methods of reference to character repertoires and their coding

Within programming languages and other methods for defining the syntax of data objects there is commonly a need to declare a specific character repertoire from among those that are specified in ISO/IEC 10646. There may also be a need to declare the corresponding coded representations applicable to that repertoire.

For any character repertoire that is in accordance with ISO/IEC 10646 a precise declaration of that repertoire should include the following parameters:

- identification of ISO/IEC 10646,
- the adopted subset of the repertoire, identified by one or more collection numbers,
- the adopted implementation level (1, 2 or 3),
- the adopted coded representation form (4-octet or 2-octet).

One of the methods now in common use for defining the syntax of data objects is Abstract Syntax Notation 1 (ASN.1) specified in ISO/IEC 8824. The corresponding coded representations are specified in ISO/IEC 8825. When this method is used the forms of the references to character repertoires and coding are as indicated in the following clauses.

N.2 Identification of ASN.1 character abstract syntaxes

The set of all character strings that can be formed from the characters of an identified repertoire in accordance with ISO/IEC 10646 is defined to be a "character abstract syntax" in the terminology of ISO/IEC 8824. For each such character abstract syntax, a corresponding object identifier value is defined to permit references to be made to that syntax when the ASN.1 notation is used.

ISO/IEC 8824-1 annex B specifies the form of object identifier values for objects that are specified in an ISO standard. In such an object identifier the features and options of ISO/IEC 10646 are identified by means of numbers (arcs) which follow the arcs "10646" and "0" which identify the whole ISO/IEC 10646.

NOTE 1 – The arc (0) is required to complement the arcs (1) and (2) which represent respectively Part 1 and Part 2 of the

previous editions of ISO/IEC 10646. These two arcs should not be used.

The first such arc following a 10646 arc identifies the adopted implementation level, and is either:

- level-1 (1), or
- level-2 (2), or
- level-3 (3).

The second such arc identifies the repertoire subset, and is either:

- all (0), or
- collections (1).

Arc (0) identifies the entire collection of characters specified in ISO/IEC 10646. No further arc follows this arc.

NOTE 2 - This collection includes private groups and planes, and is therefore not fully-defined. Its use without additional prior agreement is deprecated.

Arc (1) is followed by one or a sequence of further arcs, each of which is a collection number from annex A, in ascending numerical order. This sequence identifies the subset consisting of the collections whose numbers appear in the sequence.

NOTE 3 - As an example, the object identifier for the subset comprising the collections BASIC LATIN, LATIN-1 SUPPLEMENT, and MATHEMATICAL OPERATORS, at implementation level 1, is:

{iso standard 10646 0 level-1 (1) collections (1) 1 2 39}

ISO/IEC 8824 also specifies object descriptors corresponding to object identifier values. For each combination of arcs the corresponding object descriptors are as follows:

- 1 0 : "ISO 10646 level-1 unrestricted"
- 2 0 : "ISO 10646 level-2 unrestricted"
- 3 0 : "ISO 10646 level-3 unrestricted"

For a single collection with collection name "xxx".

- 1 1 : "ISO 10646 level-1 xxx"
- 2 1 : "ISO 10646 level-2 xxx"
- 3 1 : "ISO 10646 level-3 xxx"

For a repertoire comprising more than one collection, numbered m1, m2, etc.

- 1 1 : "ISO 10646 level-1 collections m1, m2, m3, .. "

2 1 : "ISO 10646 level-2 collections m1, m2, m3, .. "

3 1 : "ISO 10646 level-3 collections m1, m2, m3, .. "

NOTE 4 - All spaces are single spaces.

N.3 Identification of ASN.1 character transfer syntaxes

The coding method for character strings that can be formed from the characters in accordance with ISO/IEC 10646 is defined to be a "character transfer syntax" in the terminology of ISO/IEC 8824. For each such character transfer syntax, a corresponding object identifier value is defined to permit references to be made to that syntax when the ASN.1 notation is used.

In an object identifier in accordance with ISO/IEC 8824-1 annex B, the coded representation form specified in ISO/IEC 10646 is identified by means of numbers (arcs) which follow the arcs "10646" and "0" which identify the whole ISO/IEC 10646.

The first such arc is:
- transfer-syntaxes (0).

The second such arc identifies the form and is either:

- two-octet-BMP-form (2), or
- four-octet-form (4), or
- utf16-form (5), or
- utf8-form (8).

NOTE - As an example, the object identifier for the two-octet coded representation form is:

{iso standard 10646 0 transfer-syntaxes (0) two-octet-BMP-form (2)}

The following form is also valid but deprecated:

{iso standard 10646 1 transfer-syntaxes (0) two-octet-BMP-form (2)}

The corresponding object descriptors are:

- "ISO 10646 form 2"
- "ISO 10646 form 4"
- "ISO 10646 utf-16"
- "ISO 10646 utf-8".

Annex P (informative)

Additional information on characters

This annex contains additional information on some of the characters specified in clause 26 of this International Standard. This information is intended to clarify some feature of a character, such as its naming or usage, or its associated graphic symbol.

Each entry in this annex consists of the name of a character preceded by its code position in the two-octet form, followed by the related additional information. Entries are arranged in ascending sequence of code position.

When an entry for a character is included in this annex an * symbol appears immediately following its name in the corresponding table in clause 26 of this International Standard.

00AB LEFT-POINTING DOUBLE ANGLE QUOTATION MARK
This character may be used as an Arabic opening quotation mark, if it appears in a bidirectional context as described in clause 19. The graphic symbol associated with it may differ from that in the table for Row 00.

00BB RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK
This character may be used as an Arabic closing quotation mark, if it appears in a bidirectional context as described in clause 19. The graphic symbol associated with it may differ from that in the table for Row 00.

00C6 LATIN CAPITAL LETTER AE (ash)
In the first edition of this International Standard the name of this character was:
LATIN CAPITAL LIGATURE AE

00E6 LATIN SMALL LETTER AE (ash)
In the first edition of this International Standard the name of this character was:
LATIN SMALL LIGATURE AE

0189 LATIN CAPITAL LETTER AFRICAN D
This character is the capital letter form of:
0256 LATIN SMALL LETTER D WITH TAIL

019F LATIN CAPITAL LETTER O WITH MIDDLE TILDE
This character is the capital letter form of:
0275 LATIN SMALL LETTER BARRED O

01A6 LATIN LETTER YR
This character is the capital letter form of:
0280 LATIN LETTER SMALL CAPITAL R

01E2 LATIN CAPITAL LETTER AE WITH MACRON (ash)
In the first edition of this International Standard the name of this character was:

LATIN CAPITAL LIGATURE AE WITH MACRON

01E3 LATIN SMALL LETTER AE WITH MACRON (ash)
In the first edition of this International Standard the name of this character was:

LATIN SMALL LIGATURE AE WITH MACRON

01FC LATIN CAPITAL LETTER AE WITH ACUTE (ash)
In the first edition of this International Standard the name of this character was:

LATIN CAPITAL LIGATURE AE WITH ACUTE

01FD LATIN SMALL LETTER AE WITH ACUTE (ash)
In the first edition of this International Standard the name of this character was:
LATIN SMALL LIGATURE AE WITH ACUTE

0218 LATIN CAPITAL LETTER S WITH COMMA BELOW
This character is intended for use only in those cases where it is necessary to make a distinction from the letter with cedilla. Both forms of the letter may be found in a single document written in a single language, e.g. Romanian or Turkish.

In ISO/IEC 8859-2 only a single (8-bit) coded character is provided, LATIN CAPITAL LETTER S WITH CEDILLA, which maps to 015E in ISO/IEC 10646 by default, and may map by mutual agreement between sender and receiver to this letter with comma below. See ISO/IEC 8859-2 for further information on the use of that standard.

0219 LATIN SMALL LETTER S WITH COMMA BELOW
This character is intended for use only in those cases where it is necessary to make a distinction from the letter with cedilla. Both forms of the letter may be found in a single document written in a single language, e.g. Romanian or Turkish.

In ISO/IEC 8859-2 only a single (8-bit) coded character is provided, LATIN SMALL LETTER S WITH CEDILLA, which maps to 015F in ISO/IEC 10646 by default, and may map by mutual agreement between sender and receiver to this letter with comma below. See ISO/IEC 8859-2 for further information on the use of that standard.

021A LATIN CAPITAL LETTER T WITH COMMA BELOW
This character is intended for use only in those cases where it is necessary to make a distinction from the letter with cedilla. Both forms of the letter

may be found in a single document written in a single language, e.g. Romanian.

In ISO/IEC 8859-2 only a single (8-bit) coded character is provided, LATIN CAPITAL LETTER T WITH CEDILLA, which maps to 0162 in ISO/IEC 10646 by default, and may map by mutual agreement between sender and receiver to this letter with comma below. See ISO/IEC 8859-2 for further information on the use of that standard.

021B LATIN SMALL LETTER T WITH COMMA BELOW

This character is intended for use only in those cases where it is necessary to make a distinction from the letter with cedilla. Both forms of the letter may be found in a single document written in a single language, e.g. Romanian.

In ISO/IEC 8859-2 only a single (8-bit) coded character is provided, LATIN SMALL LETTER T WITH CEDILLA, which maps to 0163 in ISO/IEC 10646 by default, and may map by mutual agreement between sender and receiver to this letter with comma below. See ISO/IEC 8859-2 for further information on the use of that standard.

0280 LATIN LETTER SMALL CAPITAL R

This character is the small letter form of:
01A6 LATIN LETTER YR

03D8 GREEK LETTER ARCHAIC KOPPA

The name of this character distinguishes it from 03DE GREEK LETTER KOPPA, which is most commonly used with its numeric value, such as in the dating of legal documentation. GREEK LETTER ARCHAIC KOPPA is primarily used alphabetically to represent the letter used in early Greek inscriptions.

03D9 GREEK SMALL LETTER ARCHAIC KOPPA

The name of this character distinguishes it from 03DF GREEK SMALL LETTER KOPPA, which is most commonly used with its numeric value, such as in the dating of legal documentation. GREEK SMALL LETTER ARCHAIC KOPPA is primarily used alphabetically to represent the letter used in early Greek inscriptions.

0596 HEBREW ACCENT TIPEHA

This character may be used as a Hebrew accent tarha.

0598 HEBREW ACCENT ZARQA

This character may be used as a Hebrew accent ziyorit.

05A5 HEBREW ACCENT MERKHA

This character may be used as a Hebrew accent yored.

05A8 HEBREW ACCENT QADMA

This character may be used as a Hebrew accent azla.

05AA HEBREW ACCENT YERAH BEN YOMO

This character may be used as a Hebrew accent galgal.

05BD HEBREW POINT METEG

This character may be used as a Hebrew accent sof pasuq or siluq.

05C0 HEBREW PUNCTUATION PASEQ

This character may be used as a Hebrew accent legarme.

05C3 HEBREW PUNCTUATION SOF PASUQ

This character may be used as a Hebrew punctuation colon.

06AF ARABIC LETTER GAF

The symbol for a Hamza (see position 0633) may appear in the centre of the graphic symbol associated with this character.

06D0 ARABIC LETTER E

This character may be used as an Arabic letter Sindhi bbeh.

0F6A TIBETAN LETTER FIXED-FORM RA

This character has the same graphic symbol as that shown in the table for:

0F62 TIBETAN LETTER RA

It may be used when the graphic symbol is required to remain unchanged regardless of context.

0FAD TIBETAN SUBJOINED LETTER WA

The graphic symbol for this character occurs in two alternative forms, a full form and a short form (known as *wa.zur* (*wazur*)). The short form of the letter is shown in the table, since it occurs more frequently.

0FB1 TIBETAN SUBJOINED LETTER YA

The graphic symbol for this character occurs in two alternative forms, a full form and a short form (known as *ya.btags* (*ya ta*)). The short form of the letter is shown in the table, since it occurs more frequently.

0FB2 TIBETAN SUBJOINED LETTER RA

The graphic symbol for this character occurs in two alternative forms, a full form and a short form (known as *ra.btags* (*ra ta*)). The short form of the letter is shown in the table, since it occurs more frequently.

1100 HANGUL CHOSEONG KIYEOK ...

1112 HANGUL CHOSEONG HIEUH

The Latin letters shown in parenthesis after the names of the characters in the range 1100 to 1112 (except 110B) are transliterations of these Hangul characters. These transliterations are used in the construction of the names of the Hangul syllables that are allocated in code positions AC00 to D7A3 in this International Standard.

11A8 HANGUL JONGSEONG KIYEOK ...

11C2 HANGUL JONGSEONG HIEUH

The Latin letters shown in parenthesis after the names of the characters in the range 11A8 to 11C2 are transliterations of these Hangul characters. These transliterations are used in the construction of the names of the Hangul syllables that are allo-

cated in code positions AC00 to D7A3 in this International Standard.

17A3 KHMER INDEPENDENT VOWEL QAQ

This character is only used for Pali/Sanskrit transliteration. The use of this character is discouraged; 17A2 KHMER LETTER QA should be used instead.

17A4 KHMER INDEPENDENT VOWEL QAA

This character is only used for Pali/Sanskrit transliteration. The use of this character is discouraged; the sequence <17A2, 17B6> (KHMER LETTER QA followed by KHMER VOWEL SIGN AA) should be used instead.

17B4 KHMER VOWEL INHERENT AQ

17B5 KHMER VOWEL INHERENT AA

Khmer inherent vowels. These characters are for phonetic transcription to distinguish Indic language inherent vowels from Khmer inherent vowels. They are included solely for compatibility with particular applications; their use in other contexts is discouraged.

17D3 KHMER SIGN BATHAMASAT

This character represents a rare sign representing the first August of leap year in the lunar calendar. The use of this character is discouraged in favor of the characters from the KHMER SYMBOLS collection.

17D8 KHMER SIGN BEYYAL

This character represents the concept of ‘et cetera’. The use of this character is discouraged; other abbreviations for ‘et cetera’ also exist. The preferred spelling is the sequence <17D4, 179B, 17D4>.

234A APL FUNCTIONAL SYMBOL DOWN TACK UNDERBAR

The relation between the name of this character and the orientation of the “tack” element in its graphical symbol is inconsistent with that of other characters in this International Standard, such as:

22A4 DOWN TACK and 22A5 UP TACK

234E APL FUNCTIONAL SYMBOL DOWN TACK JOT

Information for the character at 234A applies.

2351 APL FUNCTIONAL SYMBOL UP TACK OVERBAR

Information for the character at 234A applies.

2355 APL FUNCTIONAL SYMBOL UP TACK JOT

Information for the character at 234A applies.

2361 APL FUNCTIONAL SYMBOL UP TACK DIAERESIS

Information for the character at 234A applies.

FA1F CJK COMPATIBILITY IDEOGRAPH-FA1F

This character should be considered as an extension to the block of characters CJK UNIFIED IDEOGRAPHS EXTENSION A (see clause 27). It is not a duplicate of a character already allocated in the blocks of CJK Unified Ideographs, unlike many other characters in the block CJK COMPATIBILITY IDEOGRAPHS. The source of this character, shown as described in clause 27, is:

C	J	K	V
G - Hanzi - T	Kanji	Hanja	ChuNom

𦵏

A-264B

A-0643

FA23 CJK COMPATIBILITY IDEOGRAPH-FA23

This character should be considered as an extension to the block of characters CJK UNIFIED IDEOGRAPHS EXTENSION A (see clause 27). It is not a duplicate of a character already allocated in the blocks of CJK Unified Ideographs, unlike many other characters in the block CJK COMPATIBILITY IDEOGRAPHS. The sources of this character, shown as described in clause 27, are:

C	J	K	V
G - Hanzi - T	Kanji	Hanja	ChuNom

𦵑

𦵒

F-3862

A-2728

F-2466

A-0708

FF5F FULLWIDTH LEFT WHITE PARENTHESIS

This character has a common glyph variation that looks like a double left parenthesis.

FF60 FULLWIDTH RIGHT WHITE PARENTHESIS

This character has a common glyph variation that looks like a double right parenthesis.

FFE3 FULLWIDTH MACRON

This character is the full-width form of the character: 00AF MACRON. It is also used as the full-width form of the character:

203E OVERLINE

Annex Q (informative)

Code mapping table for Hangul syllables

This annex provides a cross-reference between the Hangul syllables (and code positions) that were specified in the First Edition of this International Standard and their amended code positions as now specified in this edition.

In the First Edition of this International Standard 6656 Hangul syllables were allocated to consecutive code positions in the range 3400 to 4DFF. These Hangul syllables are now re-allocated non-consecutively to code positions in the larger range AC00 to D7A3.

The cross-reference is provided in machine-readable format that is accessible as link to this document. The content linked to is a plain text file, using ISO/IEC 646-IRV characters with LINE FEED as end of line mark, that specifies, after a 4-lines header, as many lines as Hangul syllables specified in the First Edition of this International Standard; each containing the following information organized in fixed width fields:

- 01-05 octet: First Edition code positions for Hangul syllables (hhhh)
- 06-10 octet: Current Edition code positions for Hangul syllables (hhhh).

The format definition uses 'h' as a hexadecimal unit. The space character appearing between parentheses appears as shown.

[Click on this highlighted text to access the reference file.](#)

NOTE 1 – The content is also available as a separate viewable file in the same file directory as this document. The file is named: "HangulX.txt".

NOTE 2 – The referenced files are only available to users who obtain their copy of the standard in a machine-readable format. However, the file format makes them printable.

Annex R (informative)

Names of Hangul syllables

This annex provides the names of each character in the block HANGUL SYLLABLES (AC00 - D7A3) sorted by code positions in machine-readable format that is accessible as a link to this document. The content linked to is a plain text file, using ISO/IEC 646-IRV characters with LINE FEED as end of line mark, that specifies, after a 4-lines header, all the Hangul syllable character names from ISO/IEC 10646.

These names include the syllable-names of Hangul syllables. The syllable-name is the final component of the full character name, and is derived as described in 28.3, steps 1 to 5, which is the definitive specification of the names in that block. The character names are derived as described in 28.3, steps 1 to 8.

The format of the file, after the header, is as follows:

01-04 octet: UCS-2 code position,

06 octet: TAB character,

07-end of line: character name

[Click on this highlighted text to access the reference file.](#)

NOTE 1 – The content is also available as a separate viewable file in the same file directory as this document. The file is named: "HangulSy.txt".

NOTE 2 – The referenced files are only available to users who obtain their copy of the standard in a machine-readable format. However, the file format makes them printable.

Annex S (informative)

Procedure for the unification and arrangement of CJK Ideographs

The graphic character collections of CJK unified ideographs in ISO/IEC 10646 are specified in clause 33. They are derived from many more ideographs which are found in various different national and regional standards for coded character sets (the "sources").

This Annex describes how the ideographs in this standard are derived from the sources by applying a set of unification procedures. It also describes how the ideographs in this standard are arranged in the sequence of consecutive code positions to which they are assigned.

The source references for CJK unified ideographs are specified in clause 27.1.

Within the context of ISO/IEC 10646 a unification process is applied to the ideographic characters taken from the codes in the source groups. In this process, single ideographs from two or more of the source groups are associated together, and a single code position is assigned to them in this standard. The associations are made according to a set of procedures that are described below. Ideographs that are thus associated are described here as "unified".

NOTE - The unification process does not apply to the following collections of ideographic characters:

- CJK RADICALS SUPPLEMENT (2E80 - 2EFF)
- KANGXI RADICALS (2F00 - 2FDF)
- CJK COMPATIBILITY IDEOGRAPHS (F900 - FAFF with the exception of FA0E, FA0F, FA11, FA13, FA14, FA1F, FA21, FA23, FA24, FA27, FA28 and FA29)
- CJK COMPATIBILITY IDEOGRAPHS SUPPLEMENT (2F800-2FA1F).

S.1 Unification procedure

S.1.1 Scope of unification

Ideographs that are unrelated in historical derivation (non-cognate characters) have not been unified.

Example:

士, 土

NOTE - The difference of shape between the two ideographs in the above example is in the length of the lower horizontal line. This is considered an actual difference of shape. Furthermore these ideographs have different meanings. The meaning of the first is "Soldier" and of the second is "Soil or Earth".

An association between ideographs from different sources is made here if their shapes are sufficiently similar, according to the following system of classification.

S.1.2 Two level classification

A two-level system of classification is used to differentiate (a) between abstract shapes and (b) between actual shapes determined by particular typefaces. Variant forms of an ideograph, which can not be unified, are identified based on the difference between their abstract shapes.

S.1.3 Procedure

A unification procedure is used to determine whether two ideographs have the same abstract shape or different ones. The unification procedure has two stages, applied in the following order:

- a) Analysis of component structure;
- b) Analysis of component features;

S.1.3.1 Analysis of component structure

In the first stage of the procedure the component structure of each ideograph is examined. A component of an ideograph is a geometrical combination of primitive elements. Alternative ideographs can be configured from the same set of components. Components can be combined to create a new component with a more complicated structure. An ideograph, therefore, can be defined as a component tree, where the top node is the ideograph itself, and the bottom nodes are the primitive elements. This is shown in Figure S.1.

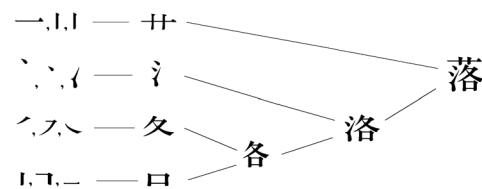


Figure S.1 - Component structure

S.1.3.2 Analysis of component features

In the second stage of the procedure, the components located at corresponding nodes of two ideographs are

compared, starting from the most superior node, as shown in Figure S.2.

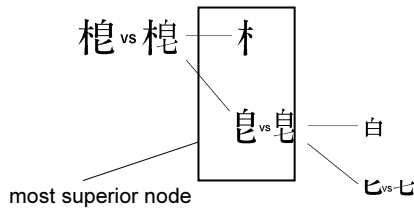


Figure S.2 - The most superior node of a component

The following features of each ideograph to be compared are examined:

- a) the number of components,
- b) the relative position of the components in each complete ideograph,
- c) the structure of corresponding components.

If one or more of the features a) to c) above are different between the ideographs in the comparison, the ideographs are considered to have different abstract shapes and are therefore not unified.

If all of the features a) to c) above are the same between the ideographs, the ideographs are considered to have the same abstract shape and are therefore unified.

S.1.4 Examples of differences of abstract shapes

To illustrate rules derived from a) to c) in S.1.3.2, some typical examples of ideographs that are not unified, owing to differences of abstract shapes, are shown below.

S.1.4.1 Different number of components

The examples below illustrate rule a) since the two ideographs in each pair have different numbers of components.

崖·厓, 肱·肱, 降·夆

S.1.4.2 Different relative positions of components

The examples below illustrate rule b). Although the two ideographs in each pair have the same number of components, the relative positions of the components are different.

峰·峯, 荊·荆

S.1.4.3 Different structure of a corresponding component

The examples below illustrate rule c). The structure of one (or more) corresponding components within the two ideographs in each pair is different.

扌·擴, 策·筵, 兴·燃, 圣·亓, ,
 僉·僉, 区·區, 夾·夾, 单·單, ,
 雀·雀, 戔·戔, 贊·贊, 襄·襄, ,
 隹·隹, 間·間, 朶·朶, 雋·雋, ,
 恒·恆, 奂·奂, 人·人, 冢·冢, ,
 爰·爰

S.1.5 Differences of actual shapes

To illustrate the classification described in S.1.2, some typical examples of ideographs that are unified are shown below. The two or three ideographs in each group below have different actual shapes, but they are considered to have the same abstract shape, and are therefore unified.

之·之·之, 示·示·示, 艮·艮·艮, 食·食·食, ,
 黃·黃, 盥·盥, 曷·曷, 包·包, ,
 青·青, 每·每, 册·册, 爭·爭, ,
 畷·畷·畷, 录·录, 步·步, 者·者, ,
 臭·臭, 并·并, 骨·骨, 呂·呂, ,
 直·直, 梟·梟, 吳·吳·吳, 眞·眞·眞, ,
 爲·為, 單·单, 曾·曾·曾, 成·成, ,
 專·專, 内·内, 晉·晋, 龜·龜, ,
 卅·卅

The differences are further classified according to the following examples.

a) Differences in rotated strokes/dots

半·半, 勺·勺, 羽·羽·羽, 酋·酋, ,
 兼·兼, 益·益

b) Differences in overshoot at the stroke initiation and/or termination

身·身, 雪·雪, 拐·拐, 不·不, ,
 非·非, 周·周, 告·告

c) Differences in contact of strokes

奧·奧, 酉·酉, 兕·兕, 查·查, 奔·奔

d) Differences in protrusion at the folded corner of strokes

巨·巨

e) Differences in bent strokes

西·西

f) Differences in folding back at the stroke termination

朱·朱

g) Differences in accent at the stroke initiation

父·父, 丈·丈, 夂·夂

h) Differences in "rooftop" modification

八·八, 宀·宀

j) Combinations of the above differences

刃·刃

These differences in actual shapes of a unified ideograph are presented in the corresponding source columns for each code position entry in the code table in clause 27 of this International Standard.

S.1.6 Source separation rule

To preserve data integrity through multiple stages of code conversion (commonly known as "round-trip integrity"), any ideographs that are separately encoded in any one of the source standards listed below have not been unified.

G-source:	GB2312-80, GB12345-90, GB7589-87*, GB7590-87*, GB8565-88*, General Purpose Hanzi List for Modern Chinese Language*
T-source:	TCA-CNS 11643-1986/1st plane, TCA-CNS 11643-1986/2nd plane, TCA-CNS 11643-1986/14th plane*
J-source:	JIS X 0208-1990, JIS X 0212-1990
K-source:	KS C 5601-1989, KS C 5657-1991

(A "*" after the reference number of a standard indicates that some of the ideographs included in that standard are not introduced into the unified collection.)

However, some ideographs encoded in two standards belonging to the same source group (e.g. GB2312-80 and

GB12345-90) have been unified during the process of collecting ideographs from the source group.

The source separation rule described in this clause only applies to the CJK UNIFIED IDEOGRAPHS block specified in the Basic Multilingual Plane.

NOTE – CJK Compatibility Ideographs are created following a rule very similar to the source separation rule. However, the end result is the combination of a single CJK Unified Ideograph and one or several CJK Compatibility Ideographs. When the source separation rule is applied, all 'similar' source CJK Ideographs result in separate CJK Unified Ideographs.

S.2 Arrangement procedure

S.2.1 Scope of arrangement

The arrangement of the CJK UNIFIED IDEOGRAPHS in the code table of clause 33 of this International Standard is based on the filing order of ideographs in the following dictionaries.

Priority	Dictionary	Edition
1	Kangxi Dictionary 康熙字典	Beijing 7th edition
2	Daikanwa Jiten 大漢和辞典	9th edition
3	Hanyu Dazidian 漢語大字典	1st edition
4	Daejajeon 大字源	1st edition

The dictionaries are used according to the priority order given in the table above. Priority 1 is highest. If an ideograph is found in one dictionary, the dictionaries of lower priority are not examined.

S.2.2 Procedure

S.2.2.1 Ideographs found in the dictionaries

a) If an ideograph is found in the Kangxi Dictionary, it is positioned in the code table in accordance with the Kangxi Dictionary order.

b) If an ideograph is not found in the Kangxi Dictionary but is found in the Daikanwa Jiten, it is given a position at the end of the radical-stroke group under which is indexed the nearest preceding Daikanwa Jiten character that also appears in the Kangxi dictionary.

c) If an ideograph is found in neither the Kangxi nor the Daikanwa, the Hanyu Dazidian and the Daejajeon dictionaries are referred to with a similar procedure.

S.2.2.2 Ideographs not found in the dictionaries

If an ideograph is not found in any of the four dictionaries, it is given a position at the end of the radical-stroke group (after the characters that are present in the dictionaries) and it is indexed under the same radical-stroke count.

S.3 Source code separation examples

The pairs (or triplets) of ideographs shown below are exceptions to the unification rules described in clause S.1 of this Annex. They are not unified because of the source separation rule described in clause S.1.6.

NOTE - The particular source group (or groups) that causes the source separation rule to apply is indicated by the letter (G, J, K, or T) that appears to the right of each pair (or triplet) of ideographs. The source groups that correspond to these letters are identified at the beginning of this Annex.

丟丟	T	兗兗	T	单单	T	国国	T
4E1F 4E22		5156 5157		5355 5358		56EF 56FD	
么么	GT	冊冊	TJ	即即	TK	圏圏	TJ
4E48 5E7A		518A 518C		5373 537D		5708 570F	
争争	GTJ	净净	G	卷卷	TJ	圓圓	T
4E89 722D		51C0 51C8		5377 5DFB		570E 5713	
仞仞	J	丸丸	T	叁叁	GT	圖圖	T
4EDE 4EED		51E2 51E3		53C1 53C2		5716 5717	
併併	T	刃刃	TJ	叁叁	T	罍罍	T
4F75 5002		5203 5204		53C3 53C4		5759 5DE0	
侶侶	T	刊刊	TJ	吕吕	T	埽埽	J
4FA3 4FB6		520A 520B		5415 5442		57D2 57D3	
俣俣	TJK	刪刪	T	吞吞	T	墜墜	T
4FC1 4FE3		5220 522A		541E 5451		5848 588D	
俞俞	T	別別	T	吳吳吳	TJ	填填	TJ
4FDE 516A		5225 522B		5433 5434 5449		5861 586B	
俱俱	T	券券	TJ	呐呐	T	增增	T
4FF1 5036		5238 52B5		5436 5450		5897 589E	
值值	T	剝剝	T	告告	T	壯壯	GTJ
5024 503C		5239 524E		543F 544A		58EE 58EF	
偷偷	T	勗勗	T	唧唧	T	壽壽	T
5077 5078		524F 5259		5527 559E		58FD 5900	
偽偽	TJ	剝剝	T	喻喻	T	夔夔	T
507D 50DE		525D 5265		55A9 55BB		5910 657B	
兌兌	T	劒劒	J	嘘嘘	T	本本	GTJ
514C 5151		5292 5294		5618 5653		5932 672C	
兔兔	TJ	勻勻	T	噓噓	GTJ	奧奧	J
514E 5154		52FB 5300		568F 5694		5965 5967	

獎獎獎	TJ	寢寢	GTJ	彈彈	T	戲戲	T
5968 596C 734E		5BDD5BE2		5F39 5F3E		622F 6231	
妝妝	GT	專專	J	亝亝	TJ	戶戶戶	T
5986 599D		5C02 5C08		5F50 5F51		6236 6237 6238	
妍妍	T	將將	GTJ	录录	T	戾戾	T
598D 59F8		5C06 5C07		5F54 5F55		623B 623E	
姍姍	T	尔尔	T	彙彙	T	拋拋	T
59CD 59D7		5C13 5C14		5F59 5F5A		629B 62CB	
姪姪	GT	尙尙	T	彝彝	J	拔拔	TJ
59EB 59EC		5C19 5C1A		5F5B 5F5C		629C 62D4	
娛娛娛	T	尙尙	T	彝彝	T	掙掙	T
5A1B 5A2F 5A31		5C2A 5C2B		5F5D 5F5E		6329 635D	
婕婕	T	檻檻	T	彥彥	T	插插插	TJ
5A55 5AAB		5C36 5C37		5F65 5F66		633F 63D2 63F7	
媮媮	T	屏屏	T	德德	T	捏捏	TJ
5A7E 5AAE		5C4F 5C5B		5FB3 5FB7		634F 63D1	
媪媪	TK	崢崢	GT	徵徵	T	搜搜	TJ
5AAA 5ABC		5CE5 5D22		5FB4 5FB5		635C 641C	
媯媯	T	巔巔	T	惠惠	TJ	揭揭	T
5AAF 5B00		5DD3 5DD4		6075 60E0		63B2 63ED	
媯媯	T	幷幷	T	悅悅	T	搖搖搖	TJ
5B0E 5B14		5E21 5E32		6085 60A6		63FA 6416 6447	
嫵嫵	GT	帶帶	TJ	悞悞	T	搵搵	T
5B24 5B37		5E2F 5E36		609E 60AE		63FE 6435	
孳孳	T	并并	T	憇憇	T	擊擊	TJ
5B73 5B76		5E76 5E77		60B3 60EA		6483 64CA	
宮宮	T	廐廐	T	愠愠	T	教教	T
5BAB 5BAE		5EC4 5ECF		6120 614D		654E 6559	
寬寬	T	弑弑	T	慎慎	TJ	斂斂	T
5BDB 5BEC		5F11 5F12		613C 614E		6553 655A	
寧寧	T	強強	T	戩戩	GT	既既	T
5BDC 5BE7		5F37 5F3A		6229 622C		65E2 65E3	

昂昂	T	歲歲	T	滂滂	T	眾眾	TJK
6602 663B		6B72 6B73		6E88 6F59		773E 884E	
晚晚	T	歿歿	T	漑漑	T	研研	T
665A 6669		6B7F 6B81		6E89 6F11		7814 784F	
暨暨	T	殼殼	GTJ	滾滾	T	祿祿	TJ
66A8 66C1		6BBB 6BBC		6EDA 6EFE		797F 7984	
曾曾	J	毀毀	T	潛潛	GTJK	禿禿	T
66FD 66FE		6BC0 6BC1		6F5B 6FF3		79BF 79C3	
楞楞	T	每每	T	瀨瀨	T	稅稅	T
67B4 67FA		6BCE 6BCF		7028 702C		7A05 7A0E	
查查	T	氳氳	T	為為	GTJ	穗穗	TJ
67E5 67FB		6C32 6C33		70BA 7232		7A42 7A57	
柵柵	T	汚汚	T	煢煢	GTJK	箏箏	GJ
67F5 6805		6C5A 6C61		712D 7162		7B5D 7B8F	
稅稅	T	沒沒	TJ	熙熙	J	箏箏	T
68B2 68C1		6C92 6CA1		7155 7199		7BB3 7C08	
榆榆	T	淨淨	TJ	媪媪	T	篡篡	T
6961 6986		6D44 6DE8		7174 7185		7BE1 7C12	
概概	T	涉涉	T	狀狀	GT	粵粵	T
6982 69EA		6D89 6E09		72B6 72C0		7CA4 7CB5	
榼榼	T	浼浼	T	瑤瑤	TJ	絕絕	T
6985 69B2		6D97 6D9A		7464 7476		7D55 7D76	
檄檄	T	淚淚	T	瓶瓶	T	綠綠	T
699D 6A27		6D99 6DDA		74F6 7501		7DA0 7DD1	
楨楨	J	淥淥	T	產產	T	緒緒	T
69C7 69D9		6DE5 6E0C		7522 7523		7DD2 7DD6	
樣樣	TJ	清清	T	瘦瘦	J	緣緣	T
69D8 6A23		6DF8 6E05		75E9 762		7DE3 7E01	
橫橫	T	渴渴	T	皞皞	T	緼緼	T
6A2A 6A6B		6E07 6E34		76A1 76A5		7DFC 7E15	
步步	T	溫溫	T	眞眞	TJ	緼緼	T
6B65 6B69		6E29 6EAB		771E 771F		7E48 7E66	

羹羹	TJ	虚虚	T	遙遙	J	頹頹	T
7FAE 7FB9		865A 865B		9059 9065		9839 983D	
翱翱	T	蛻蛻	T	邢邢	T	顏顏	TJ
7FF6 7FFA		86FB 8715		90A2 90C9		984F 9854	
胼胼	T	衛衛	TJK	郎郎	T	顛顛	J
80FC 8141		885B 885E		90CE 90DE		985A 985B	
脫脫	T	袞袞	TK	鄉鄉鄉	T	飲飲	J
812B 8131		886E 889E		90F7 9109 9115		98EE 98F2	
膾膾	T	裝裝	GJK	醞醞	T	餅餅	TJ
817D 8183		88C5 88DD		9196 919E		9905 9920	
烏烏	GT	訢訢	T	醬醬	J	馱馱	TJK
8203 8204		8A2E 8A7D		91A4 91AC		99B1 99C4	
舍舍	TJ	說說	T	鉸鉸	T	駢駢	TK
820D 820E		8AAA 8AAC		9203 9292		99E2 9A08	
舖舖	J	諫諫	TJ	銳銳	T	飭飭	T
8216 8217		8ACC 8AEB		92B3 92ED		9AA9 9AAB	
莊莊	TJ	謠謠	J	錄錄	T	高高	T
8358 838A		8B20 8B21		9304 9332		9AD8 9AD9	
菑菑	TJ	豨豨	T	鍊鍊	TK	髮髮	TJ
83D1 8458		8C5C 8C63		932C 934A		9AEA 9AEE	
盞盞	T	走走	TJ	鎮鎮	TJ	鬪鬪	T
8480 8495		8D70 8D71		93AD 93AE		9B2C 9B2D	
蔣蔣	GJ	駢駢	T	閱閱	T	鯁鯁	TJ
848B 8523		8EFF 8F27		95B1 95B2		9C1B 9C2E	
為為	T	輜輜	J	隍隍	G	鳳鳳	T
848D 853F		8F1C 8F3A		9667 9689		9CEF 9CF3	
蒞蒞	T	輻輻	T	青青	T	鶉鶉	J
8570 8580		8F3C 8F40		9751 9752		9D87 9DAB	
薰薰	T	达达	T	靜靜	GTJ	鷓鷓	J
85AB 85B0		8FBE 8FD6		9759 975C		9DC6 9DCF	
蘊蘊	T	迸迸	TJ	鞞鞞	J	麪麪	T
85F4 860A		8FF8 902C		976D 9771		9EAA 9EAB	

麼麼
9EBC 9EBD

T

黃黃
9EC3 9EC4

T

黑黑
9ED1 9ED2

T

In accordance with the unification procedures described in S.1 of this Annex the pairs (or triplets) of ideographs shown below are not unified. The reason for non-unification is indicated by the reference which appears to

the right of each pair (or triplet). For “non-cognate” see S.1.1

NOTE - The reason for non-unification in these examples is different from the source separation rule described in clause S.1.6.

冑冑
5191 80C4

non cognate

寶寶
5BF3 5BF6

S.1.4.3

胸胸
6710 80CA

non cognate

稻稻
7A32 7A3B

S.1.4.3

冲冲
51B2 6C96

S.1.4.3

廳廳
5EF0 5EF3

S.1.4.1

眺眺
6713 8101

non cognate

翱翱
7FF1 7FF6

S.1.4.3

決決
51B3 6C7A

S.1.4.3

懷懷
61D0 61F7

S.1.4.1

脞脞
6718 8127

non cognate

考考考
8007 8008 8009

S.1.4.3

況況
51B5 6CC1

S.1.4.3

斂斂
6560 656A

S.1.4.3

瞳瞳
6723 81A7

non cognate

聽聽聽
8074 807C 807D

S.1.4.1

塚塚
579B 579C

S.1.4.3

盼盼
670C 80A6

non cognate

朶朶
6735 6736

S.1.4.3

荊荊
8346 834A

S.1.4.2

孳孳
5B7C 5B7D

S.1.4.2

肫肫
670F 80D0

non cognate

灑灑
7054 7067

S.1.4.3

躲躲
8EB1 8EB2

S.1.4.3

Annex T (informative)

Language tagging using Tag Characters

The purpose of Tag characters is to associate a text attribute with a point or range of a text string. The value of a particular tag is not generally considered to be part of the content of the text. For example, tagging could be used to mark the language or the font applied to a portion of text. Outside of that usage, these characters are ignorable.

These tag characters can be used to spell out a character string in any ASCII-based tagging scheme that needs to be embedded into plain text. These characters can be easily identified by their code value and there is no overloading of usage for these tag characters. They can only express tag values and never textual content itself.

When characters are used within the context of a protocol or syntax containing explicit markup providing the same association, the Tag characters may be filtered out and ignored by these protocols.

For example, in SGML/XML context, an explicit language markup is specified. Therefore, the LANGUAGE TAG and other tag characters should not be used to mark a language in that context. The Unicode Consortium and the W3C have co-written a technical report: Unicode in XML and other Markup Languages (TR#20), available from the Unicode web site (<http://www.unicode.org>), which describes these issues in detail.

The TAGS block contains 97 dedicated tag characters consisting of a clone of the BASIC LATIN graphic characters (names formed by prefixing these BASIC LATIN names with the word 'TAG'), as well as a language tag identification character: LANGUAGE TAG and a cancel tag character: CANCEL TAG.

The tag identification character is used as a mechanism for identifying tags of different types. This enables multiple types of tags to coexist amicably embedded in plain text and solves the problem of delimitation if a tag is concatenated directly onto another tag. Although only one type of tag is currently specified, namely the language tag, the encoding of other tag identification characters in the future would allow for distinct types to be used.

T.1 Syntax for embedding tag characters

In order to embed any ASCII-derived tag in plain text, the tag is simply spelled out with the tag characters, prefixed with the relevant tag identification character. The resultant string is embedded directly in the text.

No termination character is required for a tag. A tag terminates either when the first non Special Purpose Plane character is encountered, or when the next tag identification character is encountered.

Tag arguments can only be encoded using tag characters. No other characters are valid for expressing the tag arguments.

T.2 Tag scope and nesting

The value of a tag continues from the point the tag is embedded in text until:

- either the end of the cc-data-element is reached,
- or the tag is explicitly cancelled by the CANCEL TAG character.

Tags of the same type cannot be nested. The appearance of a new embedded language tag, for example after text which was already language-tagged, simply changes the tagged value for subsequent text to that specified in the new tag.

T.3 Canceling tag values

The CANCEL TAG character is provided to allow the specific canceling of a tag value. For example to cancel a language tag, the LANGUAGE TAG must precede the CANCEL TAG character.

The usage of the CANCEL TAG character without a prefixed tag identification character cancels any tag value that may be defined.

The main function of the character is to make possible such operations as blind concatenation of strings in a tagged context without the propagation of inappropriate tag values across the string boundaries.

T.4 Language tags

Language tags are of general interest and may have a high degree of interoperability for protocol usage. For example, to embed a language tag for Japanese, the tag characters would be used as follows:

E0001 E006A E0061

The first value is the coded value of the LANGUAGE TAG character, the second corresponds to the TAG

LATIN SMALL LETTER J, and the third corresponds to the TAG LATIN SMALL LETTER A. The sequence 'ja' corresponds to the 2-letter code representing the Japanese language in ISO 639:1988.

Annex U

(informative)

Usage of musical symbols

The musical symbols repertoires are comprised of combining characters and other characters. As such their usage is specified by the clause 25. This annex describes in more details the usage of these combining characters.

U.1 Byzantine musical symbols

The Byzantine Musical Notation System makes use of the so-called 'three-stripe' effect. There are signs that appear in the Upper, Middle or Lower stripes. Other signs are known as musical characters and appear in the textual part of the notation system. Multiple signs can be stacked together in their appropriate stripe.

U.2 Western musical symbols

This international standard does not specify an encoding solution for musical scores or musical pitch. Solutions for these needs would require another description layer on top of the encoding definition of the characters specified in this standard. However, even without that additional layer, these characters can be used as simple musical reference symbols for general purposes in text descriptions of musical matters.

Extended beams are used frequently in music notation between groups of notes having short values. The format characters `MUSICAL SYMBOL BEGIN BEAM` and `MUSICAL SYMBOL END BEAM` can be used to indicate the extents of beam groupings. In some exceptional cases, beams are unclosed on one end. This can be indicated with a "null note" (`MUSICAL SYMBOL NULL NOTEHEAD`) character if no stem is to appear at the end of the beam.

Similarly, other format characters have been provided for other connecting structures. The characters

- `MUSICAL SYMBOL BEGIN TIE`
- `MUSICAL SYMBOL END TIE`
- `MUSICAL SYMBOL BEGIN SLUR`
- `MUSICAL SYMBOL END SLUR`
- `MUSICAL SYMBOL BEGIN PHRASE`
- `MUSICAL SYMBOL END PHRASE`

indicate the extent of these features.

These pairs of characters modify the layout and grouping of notes and phrases in full music notation. When musical examples are written or rendered in plain text without special software, the start/end control characters may be rendered as brackets or left un-interpreted.

More sophisticated in-line processes may interpret them, to the extent possible, in their actual control capacity, rendering ties, slurs, beams, and phrases as appropriate.

For maximum flexibility, the character set includes both pre-composed note values as well as primitives from which complete notes are constructed. Due to their ubiquity, the pre-composed versions are provided mainly for convenience.

Coding convenience notwithstanding, notes built up from alternative noteheads, stems and flags, and articulation symbols are necessary for complete implementations and complex scores. Examples of their use include American shape-note and modern percussion notations. For example,

```
MUSICAL SYMBOL SQUARE NOTEHEAD BLACK +
MUSICAL SYMBOL COMBINING STEM
```

```
MUSICAL SYMBOL X NOTEHEAD + MUSICAL SYMBOL
COMBINING STEM
```

Augmentation dots and articulation symbols may be appended to either the pre-composed or built-up notes.

In addition, augmentation dots and articulation symbols may be repeated as necessary to build a complete note symbol. For example,

```
MUSICAL SYMBOL EIGHTH NOTE + MUSICAL SYMBOL
COMBINING AUGMENTATION DOT + MUSICAL SYMBOL
COMBINING AUGMENTATION DOT + MUSICAL SYMBOL
COMBINING ACCENT
```