L2/07-218

WG2 N3275

ISO/IEC International Standard ISO/IEC 10646

Final Committee Draft

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

Technologie de l'information – Jeu universel de caractères codés ~~sur plusieurs octets~~ (JUC)

Second edition, 2008

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.ch
Web www.iso.ch

Printed in Switzerland

CONTENTS

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 of 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. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. 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 second edition of ISO/IEC 10646 cancels and replaces ISO/IEC 10646:2003. It also incorporates ISO/IEC 10646:2003 ~~/~~Amd.1:2005, ~~and ISO/IEC 10646:2003/~~Amd.2:2006, Amd.3:2007, Amd.4:2008, Amd.5:2009.

NOTE – Amendment 4 and 5 are still in progress. The text in this document is synchronized with their contents and will be updated accordingly.

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 99 000 characters from the world's scripts.

ISO/IEC 10646 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:

- CJKU_SR.txt
- CJKC_SR.txt
- IICORE.txt
- JIEx.txt
- Allnames.txt
- ~~HangulX.txt~~
- ~~HangulTb.pdf~~
- HangulSy.txt.

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 symbol.

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~~UCS codespace;
- 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 and format characters of the BMP, SMP, SIP, SSP and their coded representations within the UCS codespace;
- specifies the coded representations for control ~~functions~~characters and private use characters;
- specifies ~~the four-octet (32-bit) canonical~~three encoding forms of the UCS: ~~UCS-4~~UTF-8, UTF-16, and UTF-32;
- specifies ~~a two-octet (16-bit) BMP~~seven encoding schemes ~~form~~ of the UCS: UTF-8, UTF-16, UTF-16BE, UTF-16LE, UTF-32, UTF-32BE, and UTF-32LE~~: UCS-2~~;
- ~~specifies the coded representations for control functions;~~
- specifies the management of future additions to this coded character set.

The UCS is a encoding system different from that specified in ISO/IEC 2022. The method to designate UCS from ISO/IEC 2022 is specified in 12.2~~16.2~~.

A graphic character will be assigned only one code ~~position~~ point in the standard, located either in the BMP or in one of the supplementary planes.

NOTE – The Unicode Standard, Version 5.1 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, processing algorithms, and definitions that are useful to implementers.

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 clause~~s~~ 6~~6 and 7~~, to an identified encoding form chosen from clause 9~~13 or Annex C or Annex D~~, and to an identified encoding scheme chosen from clause 10;

b) all the graphic characters represented within that CC-data-element are taken from those within an identified subset (see 8~~12~~);

c) all the coded representations of control functions within that CC-data-element conform to clause 11~~15~~.

A claim of conformance shall identify the adopted encoding form, the adopted encoding scheme, 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 1 – The term device is defined (see 4.20) 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 encoding form(s), the adopted encoding scheme(s), and the adopted subset (by means of a list of collections and/or characters), and the selection of control functions adopted in accordance with clause 11~~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 subclauses 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 encoding form and adopted encoding scheme. As such, the originating device shall not emit ill-formed CC-data-elements.

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 encoding form and the adopted encoding scheme, and shall make any corresponding characters from the adopted subset available to the user in such a way that the user can identify them. The receiving device shall treat ill-formed CC-data-elements as an error condition and shall not interpret such data as character sequences.

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 ~~2~~ 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~~The manner in which a user is notified of either an error condition or characters not within the adopted subset is not specified by this standard.

NOTE ~~3~~ 2 – See also Annex A~~Annex J~~ for receiving devices with retransmission 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 Character Database Version 5.1 (5.0 is http://www.unicode.org/Public/5.0.0/ucd/UCD.html)

Unicode Standard Annex, UAX#9, *The Unicode Bidirectional Algorithm, Version 5.1.0, [Date TBD].*

Unicode Standard Annex, *UAX#15, Unicode Normalization Forms, Version 5.1.0, [Date TBD].*

Unicode Standard Annex, *UAX#37, Ideographic Variation Database, Version 1.0, January 2006.*

4 Terms and definitions

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

4.1
Base character
A graphic character ~~that does not graphically combine with preceding~~which is not a combining character~~s~~

NOTE – Most graphic characters are base characters. This sense of graphic combination does not preclude the presentation of base characters from adopting different contextual forms or from participating in ligatures

4.2
Basic Multilingual Plane
BMP
Plane 00 of ~~Group 00~~the UCS codespace

4.3
Block
A contiguous range of code ~~positions~~ points to which a set of characters that share common characteristics, such as a script, are allocated; ~~. A~~a block does not overlap another block; ~~. O~~one or more of the code ~~positions~~ points within a block may have no character allocated to them

4.4
Canonical ~~form~~representation
The ~~form~~ representation with which characters of this coded character set are specified using ~~four octets to represent each character~~code points within the UCS codespace

4.5
CC-data-element
coded-character-data-element
~~CC-data-element~~code unit sequence

~~4.1~~
An element of interchanged information that is specified to consist of a sequence of code units~~d~~ ~~representations of characters~~, in accordance with one or more identified standards for coded character sets; such sequence may contain code units associated with any type of code points

NOTE – Unlike previous editions of the standard, this version does not use anymore implementation levels. Its definition of CC-data-element content corresponds to the former unrestricted implementation level 3. Other definitions of CC-data-element content, previously known as level 1 and 2, are deprecated. To maintain compatibility with these previous editions, in the context of identification of coded representation in standards such as ISO/IEC 8824 and ISO/IEC 8825, the concept of implementation level may still be referenced as 'Implementation level 3'. See Annex N.

~~4.6~~
~~Cell~~
~~The place within a row at which an individual character may be allocated~~

~~4.7~~4.6
Character
A member of a set of elements used for the organization, control, or representation of textual data; ~~.~~ a character may be represented by a sequence of one or several coded characters

~~4.8~~4.7
Character boundary
Within a ~~stream of octets~~CC-data-element the demarcation between the last ~~octet of the~~code unit of a coded ~~representation of a~~ character and the first ~~octet~~ code unit of ~~that of~~ the next coded character

4.8
Code chart
Code table
A rectangular array showing the representation of coded characters allocated within a range of the UCS codespace

4.9
Coded character
An association between a character ~~together with its coded representation~~and a code point

4.10
Coded character set
A set of ~~unambiguous rules that establishes a character set and the relationship between the~~coded characters ~~of the set and their coded representation.~~

~~**4.11**~~
~~**Code chart**~~

4.11 ~~**A rectangular array showing the characters allocated to the octets in a code.**~~
Code point
Code position
Any value in the UCS codespace; the term code point is preferred

4.12
Code unit
The minimal bit combination that can represent a unit of encoded text for processing or interchange

NOTE – Examples of code units are octets (8-bit code unit) used in the UTF-8 encoding form, 16-bit code units in the UTF-16 encoding form, and 32-bit code units in the UTF-32 encoding form.

~~4.12~~**4.13**
Collection
A numbered and named set of entities;~~.~~ ~~For~~ for a non extended collection, these entities consist only of those coded characters whose code ~~positions~~ points lie within one or more identified ranges (see also 4.23~~4.21~~ for extended collection)~~.~~

NOTE – If any of the identified ranges include code ~~positions~~ points to which no character is allocated, the repertoire of the collection will change if an additional character is assigned to any of those ~~positions~~ code points 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.

~~4.13~~**4.14**
Combining character
Characters which have General Category values of Spacing Combining Mark (Mc), Non Spacing Mark (Mn), and Enclosing Mark (Me) according to the Unicode Character Database (see 3).

NOTE – These characters are ~~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 4.16~~4.15~~).

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

~~4.14~~**4.15**
Compatibility character
A graphic character included as a coded character of ISO/IEC 10646 primarily for compatibility with existing coded character sets~~.~~

~~4.15~~**4.16**
Composite sequence
A sequence of graphic characters consisting of a ~~non-combining~~base character followed by one or more combining characters, ZERO WIDTH JOINER, or ZERO WIDTH NON-JOINER (see also 4.14~~4.13~~)~~.~~

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 may be used to represent characters not encoded in ~~is not a character and therefore is not a member of~~ the repertoire of ISO/IEC 10646

~~4.16~~4.17

Control character

A control function the coded representation of which ~~consists of~~represents a single code ~~position~~point~~.~~

NOTE – Although control characters are often 'named' using terms such as DELETE, FORM FEED, ESC, these qualifiers do not correspond to formal character names. See 11~~15~~ for a list of the long names used by ISO/IEC 6429 in association with the control characters.

~~4.17~~4.18

Control function

An action that affects the recording, processing, transmission, or interpretation of data, and that is represented by a CC-data-element~~.~~

~~4.18~~4.19

Default state

The state that is assumed when no state has been explicitly specified (see F.2.1 and F.2.2)~~.~~

~~**4.19**~~

~~**Detailed code chart**~~

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

4.20

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.)

4.21

Encoding form

An encoding form determines how each UCS code point for a UCS character is to be expressed as one or more code unit used by the encoding form. ISO/IEC 10646 specifies UTF-8, UTF-16, and UTF-32

4.22

Encoding scheme

An encoding scheme specifies the serialization of the code units from the encoding form into octets

NOTE – Some of the UCS encoding schemes have the same labels as the UCS encoding form. However they are used in different context. UCS encoding forms refer to in-memory and application interface representation of textual data. UCS encoding schemes refer to octet-serialized textual data.

~~4.21~~4.23

Extended collection

A collection for which the entities can also consist of sequences of code ~~positions~~ points that are in normalization form NFC (see 21~~25~~); t~~. T~~he sequences of code ~~positions~~ points are referenced by Named UCS Sequence Identifiers (NUSI) listed in clause 1~~1~~25~~29~~ (see also 4.13~~4.12~~).

NOTE – Some collections such as 3 LATIN EXTENDED-A, 4 LATIN EXTENDED-B, 15 ARABIC EXTENDED, and many more, have the term 'extended' in their name. This does not make them extended collections

~~4.22~~4.24

Fixed collection

A collection in which every code ~~position~~ point 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~~.~~

~~4.23~~4.25

Format character

A character whose primary function is to affect the layout or processing of characters around it; i~~. I~~t generally does not have a visible representation of its own

.

4.26

General Category

GC

Value assigned to each UCS code point which determines its major class, such as letter, punctuation, and symbol; each value is defined as a two-letter abbreviation in the Unicode Character Database (see 3)

NOTE – When referred as a group containing all GC values sharing the same first letter, the group may be described using the first letter only. For example, 'L' stands for all letters 'Lu', 'Ll', 'Lt', 'Lm', and 'Lo'.

4.244.27

Graphic character

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

4.254.28

Graphic symbol

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

4.26

Group

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

4.274.29

High-half zonesurrogate code point

A code point in the range D800 to DBFF reserved for the use of 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.

4.30

High-surrogate code unit

A 16-bit code unit in the range D800 to DBFF used in UTF-16 as the leading code unit of a surrogate pair (see 9.2)

4.31

ill-formed CC-data-element

A UCS CC-data-element that purports to be in a UCS encoding form which does not conform to the specification of that encoding form (for example, an unpaired surrogate code unit is an ill-formed CC-data-element)

4.284.32

Interchange

The transfer of character coded data from one user to another, using telecommunication means or interchangeable media; interchange implies data serialization and the usage of a UCS encoding scheme.

4.294.33

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.

4.304.34

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.

~~4.31~~4.35
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.

~~4.32~~4.36
Low-~~half zone~~surrogate code point
A code point in the range DC00 to DFFF reserved for the use of~~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.~~

4.37
Low-surrogate code unit
A 16-bit code unit in the range DC00 to DFFF used in UTF-16 as the trailing code unit of a surrogate pair (see 9.2)

4.38
Mirrored character
A character whose image is mirrored horizontally in text that is laid out from right to left

~~4.33~~4.39
Octet
A 8-bit code ~~n ordered sequence of eight bits considered as a~~ unit; the value is expressed in hexadecimal notation from 00 to FF in ISO/IEC 10646 (see Annex K)~~.~~

~~4.34~~4.40
Plane
A subdivision of the UCS codespace consisting of 65536 code points. The UCS codespace contain 17 planes~~a group; of 256 x 256 cells~~.

~~4.35~~4.41
Presentation;
to present
The process of writing, printing, or displaying a graphic symbol.

~~4.36~~4.42
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~~.~~

~~4.37~~4.43
Private use plane
A plane within this coded character set; the contents of which is not specified in ISO/IEC 10646~~ (see 10)~~. Planes 0F and 10 are private use planes~~.~~

~~**4.38**~~
~~**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.~~

~~4.39~~4.44
Repertoire
A specified set of characters that are represented in a coded character set~~.~~

~~4.40~~4.45
Row
A subdivision of a plane; by multiple of 256 ~~cells~~code points~~.~~

~~4.41~~4.46
Script
A set of graphic characters used for the written form of one or more languages~~.~~

~~4.42~~4.47
Supplementary plane
A plane other than Plane 00 of ~~Group 00~~the UCS codespace; a plane that accommodates characters which have not been allocated to the Basic Multilingual Plane~~.~~

~~4.43~~4.48
Supplementary Multilingual Plane for scripts and symbols
SMP
Plane 01 of ~~Group 00~~the UCS codespace~~.~~

~~4.44~~4.49
Supplementary Ideographic Plane
SIP
Plane 02 of the UCS codespace~~Group 00.~~

~~4.45~~4.50
Supplementary Special-purpose Plane
SSP
Plane 0E of the UCS codespace~~Group 00~~

4.51
Surrogate pair
~~.~~A representation for a single character that consists of a sequence of two 16-bit code units, where the first value of the pair is a high-surrogate code unit and the second value is a low-surrogate code unit

4.52
UCS codespace
The UCS codespace consists of the integers from 0 to 10FFFF (hexadecimal) available for assigning the repertoire of the UCS characters

4.53
UCS scalar value
Any UCS code point except high-surrogate and low-surrogate code points

~~4.46~~4.54
Unpaired ~~RC-element~~surrogate code unit
A surrogate code unit ~~n RC-element~~ in a CC-data element that is either

- a high-surrogate code unit~~n RC-element from the high-half zone~~ that is not immediately followed by a low-surrogate unit~~n RC-element from the low-half zone~~, or

- a low-surrogate code unit ~~n RC-element from the low-half zone~~ that is not immediately preceded by a high-surrogate code unit~~n RC-element from the high-half zone.~~

~~4.47~~4.55
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.)

4.56
Well-formed CC-data-element
A UCS CC-data-element that purports to be in a UCS encoding form which conforms to the specification of that encoding form

~~**4.48 –**~~
~~**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 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 figure~~s~~ 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 the UCS codespace which consists of the integers from 0 to 10FFFF.~~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~~ coded characters ~~and their coded representation~~ for the following planes:

- The Basic Multilingual Plane (BMP, Plane 00~~ of Group 00~~). ~~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 00~~).

The planes from 03 to 0D are reserved for future standardization.~~Additional supplementary planes may be defined in the future to accommodate additional graphic characters.~~

The planes 0F and 10 ~~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.

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 made of 17 planes.

~~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~~

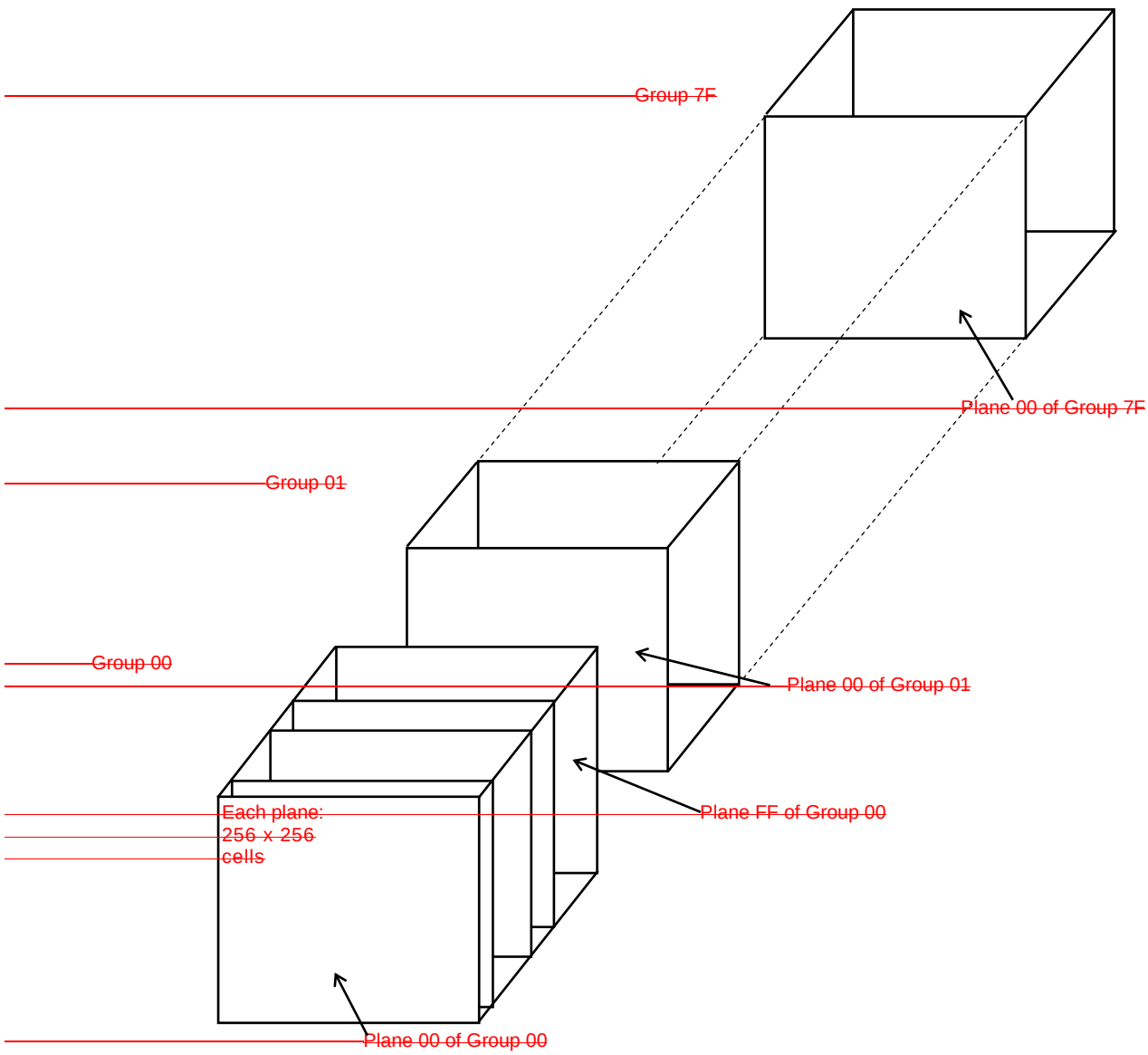


Figure 1 - Entire coding space of the Universal Multiple-Octet Coded Character Set

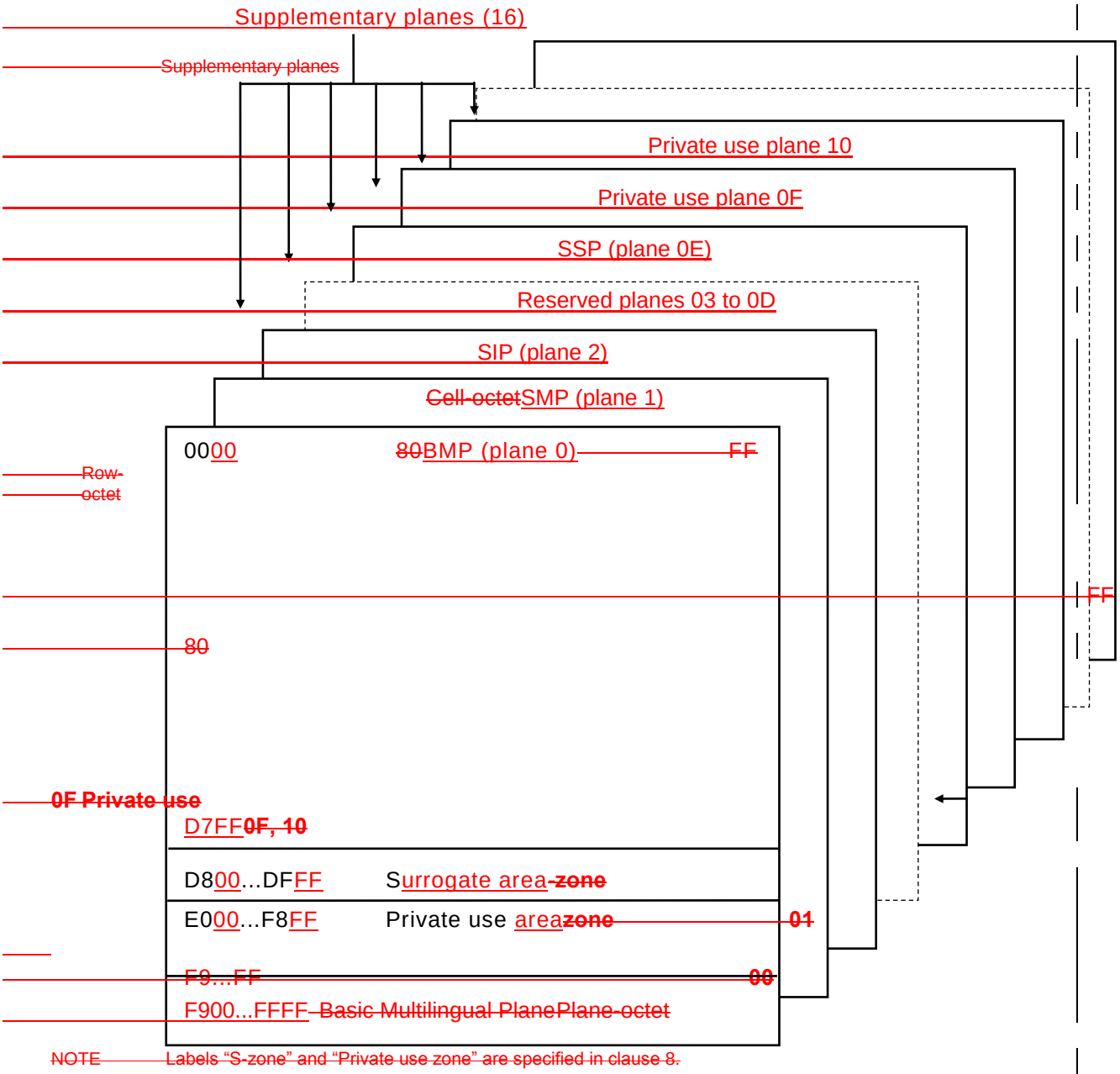


Figure 2 1 - Group 00Planes of the Universal Multiple-Octet Coded Character Set

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~~Each coded character within the UCS codespace is represented by an integer between 0 and 10FFFF identified as code point.

~~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~~	~~-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:~~When a single character is to be identified in term of its code point, it is represented by a six digit form of the integer such as

00~~0000~~ 0030for DIGIT ZERO

00~~0000~~ 0041for LATIN CAPITAL LETTER A
010000 for LINEAR B SYLLABLE B008 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 the Plane 00 (BMP), 0030 may be used to refer to DIGIT ZERO.~~

When referring to characters within plane~~s~~ 00~~ to 0F~~, the leading two ~~three~~ digits may be omitted; for characters within planes 01 to 0F, the leading digit may be omitted, such as

0030 for DIGIT ZERO
0041 for LATIN CAPITAL LETTER A
10000 for LINEAR B SYLLABLE B008 A~~. 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 Type of code points

6.3.1 Classification

UCS code points are categorized in basic types, according to their General Category value. The Table 1 summarizes the types:

Table 1: Type of code points

Basic Type	Brief Description	General Category	Character status	Code point status
Graphic	Letter, mark, number, punctuation, symbols, and spaces	L, M, N, P, S, Zs	Assigned to character	Assigned code point
Format	Invisible, but affects neighbouring characters	Cf, Zl, Zp		
Control	Control functions consisting of a single code point	Cc		
Private use	Usage defined by private agreement outside this standard	Co		

Surrogate	Permanently reserved for UTF-16	Cs	Not assigned to character	
Noncharacter	Permanently reserved for internal usage	Cn		
Reserved	Reserved for future assignment			Unassigned code point

Surrogate, noncharacter, and reserved code points are not assigned to characters and are subject to restriction in interchange. For example, surrogate code points do not have well-formed representations in any UCS encoding form.

6.3.2 Graphic characters

The same graphic character shall not be allocated to more than one code point. There are graphic characters with similar shapes in the coded character set; they are used for different purpose and have different character names.

6.3.3 Format characters

Code points 2060 to 206F, FFF0 to FFFC, and E0000 to E0FFF are reserved for Format Characters (see 16.3 and Annex F).

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

6.3.4 Control characters

Code points 0000 to 001F, 007F to 009F in the BMP are reserved for control characters (see 11).

6.3.5 Private use characters

Code points from E000 to F8FF in the BMP are reserved for private use. All code points of Plane 0F and Plane 10, except for FFFFE, FFFFF, 10FFFE, and 10FFFF are reserved for private use.

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

6.3.6 Surrogate code points

Code points D800 to DFFF are reserved for the use of the UTF-16 encoding form (see). The first half (D800 to DBFF) contains the high-surrogate code points and the second half (DC00 to DFFF) contains the low-surrogate code points.

6.3.7 Noncharacter code points

The status of noncharacter code points cannot be changed by future amendments. Noncharacters consist of FDD0-FDEF and any code point ending in the value FFFE or FFFF.

NOTE – Code point FFFE is reserved for “signature”. Code points FDD0 to FDEF, and 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 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 UTF-16.

6.3.8 Reserved code points

~~6.3~~ Reserved code points 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 points reserved for private use characters or for transformation formats.~~Octet order~~

~~In the canonical form of the coded character set, 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.~~

~~Other forms of coded representation such as UTF-16 and UTF-8, have their own sequence of octets as indicated in Annex C and Annex D respectively.~~

~~The order of octets in the coded representation form may be determined by the usage of a signature at the start of the data stream (see Annex H), by the declaration of features identification (see 16.1), or by the usage of specific transformation formats such as UTF-16BE, UTF-16LE (see Annex C), UTF-32BE, and UTF-32LE (see 13.2).~~

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 24.6~~28.6~~ for Chinese /Japanese/Korean (CJK) ideographs, or

d) follows the rule given in 24.7~~28.7~~ for Hangul syllables.

Additional rules to be used for constructing the names of characters are given in 24.2~~28.2~~.

The list of character names except for CJK ideographs and Hangul syllables is provided by the Unicode character Database in http://www.unicode.org/Public/UNIDATA/NamesList.txt with the syntax described in http://www.unicode.org/Public/UNIDATA/NamesList.html.

6.5 Short identifiers for code ~~positions~~ points (UIDs)

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

NOTE 1 – For instance, U+DC00 identifies a surrogate code ~~position~~ point that ~~is permanently reserved for UTF-16~~, and U+FFFF identifies a noncharacter code ~~position~~ point~~that is permanently reserved~~. U+0025 identifies a graphic code ~~position~~ point to which a graphic 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 six-~~eight~~-digit form of short identifier ~~shall~~ consists of the sequence of ~~eight~~ six hexadecimal digits that represents the code ~~position~~ point of the character (see 6.2~~6.2~~).

b) The four-to-~~six~~five-digit form of short identifier shall consist of the last four to ~~six~~ five digits of the ~~eight~~six-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)~~c) The character "+" (PLUS SIGN) may, as an option, precede the ~~four-to-six-~~digit form of short identifier.

~~e)~~d) The prefix letter "U" (LATIN CAPITAL LETTER U) may, as an option, precede any of the ~~four~~ three forms of short identifier defined in a~~).~~ to ~~d~~c)~~.~~ 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) ~~| {-}xxxxxxxx~~]

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~~EXAMPLE

~~As an example~~ ~~T~~the short identifier for LATIN SMALL LETTER LONG S 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 (be-fore 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-xxxxxxxx and U-xxxxxxxx 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~~ points taken from the standard. Such an identifier is known as a UCS Sequence Identifier (USI). For a sequence of n code ~~positions~~ points it has the following form:

<UID1, UID2, ..., UIDn>

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

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

6.7 Octet sequence identifiers

To represent serialized octet in the context of the encoding schemes definition (see 10), ISO/IEC 10646 defines an identifier for serialized octet sequence. For a sequence of n octets it has the following form:

$<xx_1\ xx_2\ \ldots\ xx_n>$

where xx_1, xx_2, and xx_n, represents the first, second, and n^{th} octets using two hexadecimal digits for each octet.

~~7 General requirements for the UCS~~

~~The following requirements apply to the entire coded character set.~~

~~a) 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 re-presentation of graphic characters shall be in the range 00 to 7F. On any plane, code positions FFFE and FFFF 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.~~

~~b) A "permanently reserved" code position cannot be changed by future amendments.~~

~~c) 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 for-mats.~~

~~d) 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~~

~~The Plane 00 of Group 00 is 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 to 001F, 007F to 009F in the BMP are reserved for control characters (see 15).~~

~~Code positions 2060 to 206F, FFF0 to FFFC, and E0000 to E0FFF are reserved for Format Characters (see Annex F).~~

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

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

~~Code positions E000 to F8FF are reserved for private use (see 10.1). These positions are known as the private use zone.~~

~~In addition to code positions FFFE and FFFF (see 7a)), code positions FDEF to FDD0 are also permanently reserved.~~

~~NOTE 3 – Code position FFFE is reserved for "signature" (see annex H). Code positions FDD0 to FDEF, and 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 signalling end-of-text. Furthermore, since 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.~~

~~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 are 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 10.2 for the definition of Plane 0F and 10 of Group 00.~~

~~NOTE 1 – 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 2 – 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.~~

~~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 permanently reserved.~~

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

~~10 Private use planes~~

~~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~~7 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~~8 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~~8.1 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 inter-work 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~~ points as defined in ISO/IEC 10646.

~~12.2~~8.2 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~~ code points from 0020 to 007E ~~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~~9 ~~Coded representation forms of the~~ UCS encoding forms

ISO/IEC 10646 provides three encoding forms expressing each UCS scalar value in a unique sequence of one or more code units. These are named UTF-8, UTF-16, and UTF-32 respectively.

9.1 UTF-8

UTF-8 is the UCS encoding form that assigns each UCS scalar value to an octet sequence of one to four octets, as specified in table 2.

- 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 code points from 0000 to 001F, and the control character in code point 007F, are represented without the padding octets specified in clause 11, 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-sequences 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 code unit representation of that character.

Table 2 specifies the bit distribution for the UTF-8 encoding form, showing the ranges of UCS scalar values corresponding to one, two, three, and four octet sequences.

Table 2: UTF-8 Bit distribution

Scalar value	1st octet	2nd octet	3rd octet	4th octet
00000000 0xxxxxxx	0xxxxxxx			
00000yyy yyxxxxxx	110yyyyy	10xxxxxx		
zzzzyyyy yyxxxxx	1110zzzz	10yyyyyy	10xxxxxx	
000uuuuu zzzzyyyy yyxxxxxx	11110uuu	10uuzzzz	10yyyyyy	10xxxxxx

Because surrogate code points are not UCS scalar values, any UTF-8 sequence that would otherwise map to code points D800-DFFF is ill-formed.

Table 3 lists all the ranges (inclusive) of the octet sequences that are well-formed in UTF-8. Any UTF-8 sequence that does not match the patterns listed in table 3 is ill-formed

Table 3: Well-formed UTF-8 Octet sequences

Code points	1st octet	2nd octet	3rd octet	4th octet
0000-007F	00-7F			
0080-07FF	C2-DF	80-BF		
0800-0FFF	E0	A0-BF	80-BF	
1000-CFFF	E1-EC	80-BF	80-BF	
D000-D7FF	ED	80-9F	80-BF	
E000-FFFF	EE-EF	80-BF	80-BF	
10000-3FFFF	F0	90-BF	80-BF	80-BF
40000-FFFFF	F1-F3	80-BF	80-BF	80-BF
100000-10FFFF	F4	80-8F	80-BF	80-BF

As a consequence of the well-formedness conditions specified in table 9.2, the following octet values are disallowed in UTF-8: C0-C1, F5-FE

9.2 UTF-16

UTF-16 is the UCS encoding form that assigns each UCS scalar value to a sequence of one to two unsigned 16-bit code units, as specified in table 4.

In the UTF-16 encoding form, code points in the range 0000-D7FF and E000-FFFF are represented as a single 16-bit code unit; code points in the range 10000-10FFFF are represented as pairs of 16-bit code units. These pairs of special code units are known as surrogate pairs.

The values of the code units used for surrogate pairs are disjoint from the code units used for the single code unit representation, thus maintaining non-overlap for all code point representations in UTF-16.

UTF-16 optimizes the representation of characters in the BMP which contains the vast majority of common use characters.

Because surrogate code points are not UCS scalar values, unpaired surrogate code units are ill-formed.

Table 4 specifies the bit distribution for the UTF-16 encoding form. Calculation of the surrogate pair values involves subtraction of 10000 to account for the starting offset to the scalar value (expressed as 'wwww = uuuuu-1' in the table).

Table 4: UTF-16 Bit distribution

Scalar value	UTF-16
xxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxx
000uuuuuxxxxxxxxxxxxxxxx	110110wwwwxxxxxx 110111xxxxxxxxxx

NOTE – Former editions of this standard included references to a two-octet BMP form called UCS-2 which would be a subset of the UTF-16 encoding form restricted to the BMP UCS scalar values. The UCS-2 form is deprecated.

9.3 UTF-32 (UCS-4)

UTF-32 (or UCS-4) is the UCS encoding form that assigns each UCS scalar value to a single unsigned 32-bit code unit. The terms UTF-32 and UCS-4 can be used interchangeably to designate this encoding form.

Because surrogate code points are not UCS scalar values, UTF-32 code units in the range 0000 D800-0000 DFFF are ill-formed.

10 UCS Encoding schemes

Encoding schemes are octet serialization specific to each UCS encoding form, including the specification of a signature, if allowed. The signature is the code unit sequence corresponding to the code point FEFF ZERO WIDTH NO-BREAK SPACE in the corresponding encoding form. When used, a signature at the beginning of a stream of serialized octets indicates the order of the octets within the encoding form used for the representation of the characters.

ISO/IEC 10646 specifies seven encoding schemes: UTF-8, UTF-16BE, UTF-16LE, UTF-16, UTF-32BE, UTF-32LE, and UTF-32.

10.1 UTF-8

The UTF-8 encoding scheme serializes a UTF-8 code unit sequence in exactly the same order as the code unit sequence itself.

When represented in UTF-8, the signature turns into the octet sequence <EF BB BF>. Its usage at the beginning of a UTF-8 data stream is neither required or recommended but does not affect conformance

10.2 UTF-16BE.

The UTF-16BE encoding scheme serializes a UTF-16 CC-data-element by ordering octets in a way that the more significant octet precedes the less significant octet (also known as big-endian ordering).

In UTF-16BE, an initial octet sequence of <FE FF> is interpreted as FEFF ZERO WIDTH NO-BREAK SPACE and does not convey a signature meaning.

10.3 UTF-16LE

The UTF-16LE encoding scheme serializes a UTF-16 CC-data-element by ordering octets in a way that the less significant octet precedes the more significant octet (also known as little-endian ordering).

In UTF-16LE, an initial octet sequence of <FF FE> is interpreted as FEFF ZERO WIDTH NO-BREAK SPACE and does not convey a signature meaning.

10.4 UTF-16

The UTF-16 encoding scheme serializes a UTF-16 CC-data-element by ordering octets in a way that either the less significant octet precedes or follows the more significant octet.

In the UTF-16 encoding scheme, the initial signature read as <FE FF> indicates that the more significant octet precedes the less significant octet, and <FF FE> the reverse. The signature is not part of the textual data.

In the absence of signature, the octet order of the UTF-16 encoding scheme is that the more significant octet precedes the less significant octet.

10.5 UTF-32BE

The UTF-32BE encoding scheme serializes a UTF-32 CC-data-element by ordering octets in a way that the more significant octets precede the less significant octets (also known as big-endian ordering).

In UTF-32BE, an initial octet sequence of <00 00 FE FF> is interpreted as FEFF ZERO WIDTH NO-BREAK SPACE and does not convey a signature meaning.

10.6 UTF-32LE

The UTF-32LE encoding scheme serializes a UTF-32 CC-data-element by ordering octets in a way that the less significant octets precede the more significant octets (also known as little-endian ordering).

In UTF-32LE, an initial octet sequence of <FF FE 00 00> is interpreted as FEFF ZERO WIDTH NO-BREAK SPACE and does not convey a signature meaning.

10.7 UTF-32

The UTF-32 encoding scheme serializes a UTF-32 code unit sequence by ordering octets in a way that either the less significant octet precedes or follows the more significant octet.

In the absence of signature, the octet order of the UTF-32 encoding scheme is that the more significant octets precede the less significant octets.

~~eight alternative forms of coded representation of characters. Four of these forms are specified in this clause (UCS-2, UCS-4, UTF-32BE, and UTF-32LE). Three others are specified in Annex C (UTF16, UTF-16BE, and UTF-16LE). Finally, UTF-8 is specified in Annex D.~~

~~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 (UCS-2)~~

~~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). For serialization purpose, a signature may be used (see Annex H).~~

~~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 forms (UCS-4, UTF-32BE, and UTF-32LE)~~

~~These canonical forms permit 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 UCS-4, 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.~~

~~UCS-4 is also referred to as UCS Transformation Format (UTF-32). For serialization purpose, a signature may be used (see Annex H).~~

~~NOTE 2 – UTF-32 was originally specified by the Unicode Standard and restricted to the code positions in Planes 00 to 10 (U+0000 to U+10FFF). Because code positions in all other planes are now permanently reserved, UCS-4 and UTF-32 can be used interchangeably for all assigned characters.~~

~~Two additional four-octet UCS Transformation Formats are specified for serialization purpose.~~

~~1) UTF-32BE: in the ordering of octets the more significant octets precede the less significant octets, as specified in 6.2, and no signatures appear;~~

~~2) UTF-32LE: in the ordering of octets the less significant octets precede the more significant octets, and no signatures appear.~~

~~14 CC-data-element content~~

~~A CC-data element may contain coded representations of any characters.~~

~~NOTE – Unlike previous editions of the standard, this version does not use anymore implementation levels. Its definition of CC-data-element content corresponds to the former implementation level 3. Other definitions of CC-data-element content, previously known as level 1 and 2, are deprecated. To maintain compatibility with these previous editions, in the context of identification of coded representation in standards such as ISO/IEC 8824 and ISO/IEC 8825, the concept of implementation level may still be referenced as 'Implementation level 3'. See Annex N.~~

~~15~~11 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 code unit of the adopted encoded form (see 9~~13, Annex C, and Annex 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~~UTF-16 encoding form, and "0000 000C" in the ~~four-octet~~UTF-32 encoding 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 "1B 20 40" in the UTF-8 encoding form, by "001B 0020 0040" in the ~~two-octet~~UTF-16 encoding form, and "0000 001B 0000 0020 0000 0040" in the ~~four-octet~~UTF-32 encoding form.

NOTE 1 – 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.

NOTE 2 – The following list provides the long names from ISO/IEC 6429 used in association with the control characters.

0000 NULL
0001 START OF HEADING
0002 START OF TEXT
0003 END OF TEXT
0004 END OF TRANSMISSION
0005 ENQUIRY
0006 ACKNOWLEDGE
0007 BELL
0008 BACKSPACE
0009 CHARACTER TABULATION
000A LINE FEED
000B LINE TABULATION
000C FORM FEED
000D CARRIAGE RETURN
000E SHIFT-OUT
000F SHIFT-IN
0010 DATA LINK ESCAPE
0011 DEVICE CONTROL ONE
0012 DEVICE CONTROL TWO
0013 DEVICE CONTROL THREE
0014 DEVICE CONTROL FOUR
0015 NEGATIVE ACKNOWLEDGE
0016 SYNCHRONOUS IDLE
0017 END OF TRANSMISSION BLOCK

0018 CANCEL
0019 END OF MEDIUM
001A SUBSTITUTE
001B ESCAPE
001C INFORMATION SEPARATOR FOUR
001D INFORMATION SEPARATOR THREE
001E INFORMATION SEPARATOR TWO
001F INFORMATION SEPARATOR ONE
007F DELETE
0082 BREAK PERMITTED HERE
0083 NO BREAK HERE
0084 INDEX
0085 NEXT LINE
0086 START OF SELECTED AREA
0087 END OF SELECTED AREA
0088 CHARACTER TABULATION SET
0089 CHARACTER TABULATION WITH JUSTIFICATION
008A LINE TABULATION SET
008B PARTIAL LINE FORWARD
008C PARTIAL LINE BACKWARD
008D REVERSE LINE FEED
008E SINGLE-SHIFT TWO
008F SINGLE-SHIFT THREE
0090 DEVICE CONTROL STRING
0091 PRIVATE USE ONE
0092 PRIVATE USE TWO
0093 SET TRANSMIT STATE
0094 CANCEL CHARACTER
0095 MESSAGE WAITING
0096 START OF GUARDED AREA
0097 END OF GUARDED AREA
0098 START OF STRING
009A SINGLE CHARACTER INTRODUCER
009B CONTROL SEQUENCE INTRODUCER
009C STRING TERMINATOR
009D OPERATING SYSTEM COMMAND
009E PRIVACY MESSAGE
009F APPLICATION PROGRAM COMMAND

The control character 0084 INDEX has been removed from ISO/IEC 6492:1992. In addition, the control characters 000E and 000F are named SHIFT-OUT and SHIFT-IN respectively in 7-bit environment and LOCKING-SHIFT ONE and LOCKING-SHIFT ZERO respectively in 8-bit environment.

~~16~~12 Declaration of identification of features

~~16.1~~12.1 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 encoding form and the encoding scheme) 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 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.

In the context of these identifications, because the more significant octets shall precede the less significant octets when serialized, the only encoding schemes that can be selected are UTF-8, UTF-16BE, and UTF-32BE according to the relevant encoding forms (UTF-8, UTF-16, and UTF-32 respectively).

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~~12.2 Identification of a UCS encoding ~~coded representation~~ form

When the escape sequences from ISO/IEC 2022 are used, the identification of a ~~coded representation form of~~ UCS encoding form (see 9~~13~~) specified by ISO/IEC 10646 shall be by a designation sequence chosen from the following list:

ESC 02/05 02/15 04/09~~5~~
UTF-8 encoding form; UTF-8 encoding scheme

ESC 02/05 02/15 04/12
UTF-16 encoding form; UTF-16BE encoding scheme

~~CS-2~~

ESC 02/05 02/15 04/06
~~UCS-4~~UTF-32 encoding form; UTF-32BE encoding scheme

~~or from the lists in C.5 for UTF-16 forms and D.6 for UTF-8 forms.~~

NOTE 1 – The following designation sequences: ESC 02/05 02/15 04/00, ESC 02/05 02/15 04/01, ESC 02/05 02/15 04/03, ESC 02/05 02/15 04/04, 02/05 02/15 04/07, 02/05 02/15 04/08, 02/05 02/15 04/10, 02/05 02/15 04/011 used in previous versions of this standard to identify implementation levels 1 and 2 are deprecated. The remaining designation sequences correspond to the former level 3 which is now the only supported CC-data-element content definition.

NOTE 2 – The following escape sequence may also be used:

ESC 02/05 04/07
UTF-8 encoding form; UTF-8 encoding scheme

The escape sequence used for a return to the coding system of ISO/IEC 2022 is not padded (see 12.5).

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 11~~15~~.

~~16.3~~12.3 Identification of subsets of graphic characters

When the control sequences of ISO/IEC 6429 are used, the identification of subsets (see 8~~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~~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 11~~15~~.

~~16.4~~12.4 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 11~~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 other 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 11~~15~~.

~~16.5~~12.5 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 11~~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 clause 12.2~~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~~13 Structure of the code tables and lists

Clause 30~~34~~ sets out the detailed code ~~tables~~ charts and the lists of character names for the graphic characters. It specifies graphic characters, their coded representation, and the character name for each character.

NOTE – Clause 30~~34~~ also includes additional information on characters clarifying some feature of a character, such as its naming or usage, or its associated graphic symbol.

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~~ charts 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~~chart.

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~~ chart~~rows~~ for the relevant script and for symbols and digits. In addition, annex G contains an alphabetically sorted list of character names.

~~18~~14 Block and collection names

~~18.1~~14.1 Block names

Named blocks of contiguous code ~~positions~~ points 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~~A.2~~, and are illustrated in figures ~~3~~ 2 to ~~7~~6.

Rules to be used for constructing the names of blocks are given in 24.4.1~~28.4.1~~.

~~18.2~~14.2 Collection names

Collections are shown in Annex A~~Annex A~~.

Rules to be used for constructing the names of collections are given in 24.4.2~~28.4.2~~.

~~19~~15 Mirrored characters in bidirectional context

~~19.1~~15.1 Mirrored characters

A class of characters has special significance in the context of bidirectional text. 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~~ is in effect at the point in the CC-data-element where the coded representation of the character appears. The list of these characters is ~~provided in Annex E~~determined by having the 'Bidi_Mirrored' property set to 'Y' in the Unicode Character Database (see 3).

NOTE 1 – ~~That list also represents all characters which have the 'Bidi Mirrored' property in the Unicode Standard~~Typically, a mirrored character has its image mirrored horizontally in text that is laid out from right to left. However, for some mathematical symbols, the 'mirrored' form is not an exact mirror image. See the Unicode Technical Report #25, "Unicode Support for Mathematics" for additional details.

~~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.~~

This character mirroring is not limited to paired characters and shall be applied to all characters belonging to that class.

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.

NOTE 2 – Many ancient scripts and some scripts in modern use can be written either right-to-left or left-to-right. It is often customary for one of these scripts to use the appropriately mirrored graphical symbol for any character represented by a graphic symbol that is not symmetric around the vertical axis. In such cases, it is up to the rendering system to display the graphic image appropriate for the writing direction employed. The directionality of the representative graphic symbol shown in the character code charts matches the default writing direction for the script.

Examples of such scripts include, but are not limited to, Old Italic, an ancient script for which the default writing direction in this standard is left-to-right, and Cypriot, an ancient script for which the default writing direction in this standard is right-to-left.

~~19.2~~15.2 Directionality of bidirectional text

The Unicode Bidirectional Algorithm (see 3~~3~~) describes the algorithm used to determine the directionality for bidirectional text.

~~20~~16 Special characters

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

~~20.1~~16.1 Space characters

The following characters are space characters. They represent all characters which have~~are~~ the General Category value set to 'Zs'.

Code Point ~~Position~~	Name
0020	SPACE
00A0	NO-BREAK SPACE
1680	OGHAM SPACE MARK
180E	MONGOLIAN VOWEL SEPARATOR
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
202F	NARROW NO-BREAK SPACE
205F	MEDIUM MATHEMATICAL SPACE
3000	IDEOGRAPHIC SPACE

~~20.2~~16.2 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~~16.3 Format Characters

The following characters are format characters (see 6.3.3). They represent all characters which have the General Category value set to 'Cf', 'Zl', and 'Zp'. See~~also~~ Annex F~~Annex F)~~.

Code Point	Name
00AD	SOFT HYPHEN
~~034F~~	~~COMBINING GRAPHEME JOINER~~
0600	ARABIC NUMBER SIGN
0601	ARABIC SIGN SANAH
0602	ARABIC FOOTNOTE MARKER
0603	ARABIC SIGN SAFHA
06DD	ARABIC END OF AYAH
070F	SYRIAC ABBREVIATION MARK
17B4	KHMER VOWEL INHERENT AQ
17B5	KHMER VOWEL INHERENT AA
~~180E~~	~~MONGOLIAN VOWEL SEPARATOR~~
1A60	LANNA SIGN SAKOT
1CBF	MEITEI MAYEK SIGN VIRAMA
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~~
2060	WORD JOINER
2061	FUNCTION APPLICATION
2062	INVISIBLE TIMES
2063	INVISIBLE SEPARATOR
2064	INVISIBLE PLUS
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
~~10A3F~~	~~KHAROSHTHI VIRAMA~~
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
E0001	LANGUAGE TAG
E0020-E007F	TAG SPACE to CANCEL TAG

16.4 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 with this International Standard. The annex ? describes them in more details. The list of IDC follows:

Code Point	Name
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

~~20.4~~16.5 Variation selectors and variation sequences

Variation selectors are a specific class of combining characters immediately following a non decomposable base character and which indicate a specific variant form of graphic symbol for that character. A decomposable character is a character for which there exists an equivalent composite sequence. The character sequence consisting of a non decomposable base character followed by a variation selector is called a variation sequence.

NOTE 1 – 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.

Only the variation sequences defined or referenced in this clause indicate a specific variant form of graphic symbol; all other such sequences are undefined. Furthermore, variation selectors following other base characters and any non-base characters have no effect on the selection of the graphic symbol for that character.

No variation sequences using characters from VARIATION SELECTOR-2 to VARIATION SELECTOR-16 are defined at this time. Variations sequences composed of a unified ideograph as the base character and one of VARIATION SELECTOR-17 to VARIATION SELECTOR-256 from the Supplementary Special-purpose Plane (SSP) are registered in the Ideographic Variation Database defined by Unicode Technical Standard #37.

NOTE 2 – The Ideographic Variation Database is currently empty. When entries are registered, these variation sequences will be referenced by this standard.

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

NOTE 3 – 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 OR LESS-THAN with similar following the slant of the upper leg
<2A9E, FE00>	SIMILAR OR GREATER-THAN with similar following the slant of the upper leg
<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 list 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.

NOTE 4 – 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.

Sequence (UID notation)	position	Description of variant appearance
<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 second form
<1874, 180D>	medial	MONGOLIAN LETTER MANCHU KA feminine second form
<1876, 180B>	initial, medial	MONGOLIAN LETTER MANCHU FA second form
<1880, 180B>	all	MONGOLIAN LETTER ALI GALI ANUSVARA ONE second form
<1881, 180B>	all	MONGOLIAN LETTER ALI GALI VISARGA ONE second 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

The following list provides a description of the variant appearances corresponding to the use of appropriate variation selectors with all allowed base Phags-pa characters. These variation selector sequences do not select fixed visual representation; rather, they select a representation that is reversed from the normal form predicted by the preceding character.

Sequence (UID notation) **Description of variant appearance**

<A856, FE00>	PHAGS-PA LETTER reversed shaping SMALL A
<A85C, FE00>	PHAGS-PA LETTER reversed shaping HA
<A85E, FE00>	PHAGS-PA LETTER reversed shaping I
<A85F, FE00>	PHAGS-PA LETTER reversed shaping U
<A860, FE00>	PHAGS-PA LETTER reversed shaping E
<A868, FE00>	PHAGS-PA SUBJOINED LETTER reversed shaping YA

NOTE 5 – The variation selector only selects a different *appearance* of an already encoded character. It is not intended as a general code extension mechanism.

NOTE 6 – The exhaustive list of standardized variants is also described as *StandardizedVariants.html* in the Unicode character database (http://www.unicode.org/Public/5.0.0/ucd/StandardizedVariants.html).

~~20.5 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~~17 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 code points~~sitions~~ within~~in~~ rows from FB to FF.

~~22~~18 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 code point~~sition~~s within rows F9, FA, FE, and FF, and within rows 31 and 33. Some compatibility characters are also allocated within other rows.

NOTE 1 – There are twelve code ~~positions~~ points in the row FA of the BMP which are allocated to CJK Unified Ideographs.

Within the Supplementary Ideographic Plane (SIP) these characters are allocated to code point~~sition~~s within rows F8 to FA.

The CJK compatibility ideographs are ideographs that should have been unified with one of the CJK unified ideographs, per the unification rule described in annex S. 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~~ points for them.

NOTE 2 – 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~~19 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~~20 Combining characters

This clause specifies the use of combining characters (see 4.14). ~~A list of combining characters is shown in Annex B.~~

~~NOTE – The names of many script-independent combining characters contain the word "COMBINING".~~

~~24.1~~20.1 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 00AD NO-BREAK SPACE. For example, grave accent can be composed as 00AD NO-BREAK SPACE followed by 0300 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.

~~24.2~~20.2 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 to show their position relative to the base character. 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".

NOTE – Diacritics are the principal class of combining characters used in European alphabets. For many other scripts used in India and South East Asia, combining characters encode vowel letters; as such they are not generally referred to as "diacritical marks".

~~24.3~~20.3 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 (see 21~~25~~) these coded representations ~~creates a unique~~reduces significantly, but does not eliminate, the occurrences of these multiple representations.

NOTE – For example, 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 normalization forms are applied on those alternate coded representations, only one representation remains. The form of the remaining representation depends on the normalization form used.

~~24.4~~20.4 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:

~~e)~~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 re-presentations 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 0E34 to 0E37 and, above that, one of four tone marks 0E48 to 0E4B. The order of the coded representation is: base consonant, followed by a vowel, followed by a tone mark.

~~a)~~b) Some specific combining characters override the default stacking behaviour 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 behaviour are associated with specific scripts or alphabets. For example, the COMBINING GREEK KORONIS (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.

~~b)~~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.

~~24.5~~20.5 Collections containing combining characters

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

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

20.6 Combining Grapheme Joiner

The character 034F COMBINING GRAPHEME JOINER is used to indicate that adjacent characters are to be treated as a unit for the purpose of language-sensitive collation and searching. In language-sensitive collation and searching, the combining grapheme joiner should be ignored unless it specifically occurs with a tailored collation element mapping. For rendering, the combining grapheme joiner is invisible.

NOTE 1 – The combining grapheme joiner may be used to differentiate two usages of a combining character by using it for one of the two cases. For example, where a distinction is needed between the German umlaut and the tréma, the COMBINING GRAPHEME JOINER (034F) followed by the COMBINING DIAERESIS (0308) should be used to represent the tréma while the COMBINING DIAERESIS (0308) alone should be used to represent the German umlaut.

~~25~~21 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 (see 3~~3~~). There are four normalization forms:

1) Normalization Form D (NFD) which is a canonical decomposition,
2) Normalization Form C (NFC) which is a canonical decomposition followed by canonical composition,
3) Normalization Form KD (NFKD) which is a compatibility decomposition,
4) Normalization Form KC (NFKC) which is a compatibility decomposition followed by canonical composition.

NOTE 1 – The result of applying any of these normalization forms onto a CC-data-element is intended to stay stable over time. It means that the normalized representation of a CC-data-element consisting of characters assigned in this version of the standard remains normalized even when the standard is amended.

NOTE 2 – Some normalization forms ~~favor~~favour composite sequences over shorter representations of text, others ~~favor~~favour 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. The union of their repertoire is identical to the fixed collection UNICODE 3.2 (see A.6.2~~A.6.2~~).

NOTE 3 – The goal of normalization is to provide a unique normalized result for any given CC-data element to facilitate, among other things, identity matching. A normalized form does not necessarily represent the optimal sequence from a linguistic point of view.

~~26~~22 Special features of individual scripts and symbol repertoires

~~26.1~~22.1 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), Jungseong (syllable-peak), and Jongseong (syllable-final). 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 (115F). An incomplete syllable composed of a Choseong alone shall be followed by a JUNGSEONG FILLER (1160).

NOTE 1 – Hangul Jamo are not combining characters.

NOTE 2 – When a combining character such as HANGUL SINGLE DOT TONE MARK (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~~22.2 Features of scripts used in India and some other South Asian countries

In the code charts for Rows 09 to 0D and 0F, and for the MYANMAR block in Row 10, of the BMP (see 30~~34~~) 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.

EXAMPLE 1 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

EXAMPLE 2 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 (see 4.16~~4.15~~).

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.

~~26.3~~22.3 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.

~~27~~23 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.

The source reference information establishes the character identity for CJK Ideographs. A source reference is established by associating a CJK Ideograph code ~~position~~ point with one or several values in the source standards listed in 23.1~~27.1~~ and 23.4~~27.4~~. Such a source standard originates from the following categories:

- Hanzi G sources,
- Hanzi H sources,
- Hanzi M sources,
- Hanzi T sources,
- Kanji J sources,
- Hanja K sources,
- Hanja KP sources,
- ChuNom V sources, and
- Unicode U sources

For a given code ~~position~~point, only one source reference can be created for each of the source standard category (G, H, M, T, J, K, KP, V, and U). In order to provide a comprehensive coverage for a source standard category, when a source standard is referenced, all its unique associations with existing CJK Ideographs are documented.

~~27.1~~23.1 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 charts in 30~~34~~, are described in Annex S~~Annex S~~.

NOTE 1 – The source separation rule described by the clause S.1.6~~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 the BMP and the SIP. The current full set of CJK Unified Ideographs is represented by the collection 385 CJK UNIFIED IDEOGRAPHS-2008 (See A.1~~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
G9	GB18030-2000
GE	GB16500-95
G_4K	Siku Quanshu（四庫全書）
G_BK	Chinese Encyclopedia（中國大百科全書）
G_CH	Ci Hai（辞海）
G_CY	Ci Yuan（辭源）
G_CYY	Chinese Academy of Surveying and Mapping Ideographs (中国测绘科学院用字)
G_FZ	Founder Press System（方正排版系统）
G_GH	Gudai Hanyu Cidian (古代汉语词典)
G_GJZ	Commercial Press Ideographs (商务印书馆用字)
G_HC	Hanyu Dacidian（漢語大詞典）
G_HZ	Hanyu Dazidian ideographs（漢語大字典）
G_KX	Kangxi Dictionary ideographs（康熙字典） including the addendum（康熙字典）補遺
G_XC	Xiandai Hanyu Cidian (现代汉语词典)
G_ZFY	Hanyu Fangyan Dacidian (汉语方言大辞典)
G_ZJW	Yinzhou Jinwen Jicheng Yinde (殷周金文集成引得)

The Hanzi H source is

H	Hong Kong Supplementary Character Set – 2004

The Hanzi M source is

MAC	Macao Information System Character Set (澳門資訊系統字集)

The Hanzi T sources are

T1	TCA-CNS 11643-1992 1st plane
T2	TCA-CNS 11643-1992 2nd plane
T3	TCA-CNS 11643-1992 3rd plane with some additional characters
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
TC	TCA-CNS 11643-1992 12th plane
TD	TCA-CNS 11643-1992 13th plane
TE	TCA-CNS 11643-1992 14th plane
TF	TCA-CNS 11643-1992 15th plane

The Kanji J sources are

J0	JIS X 0208-1990
J1	JIS X 0212-1990
J3	JIS X 0213:2000 level-3
J3A	JIS X 0213:2004 level-3
J4	JIS X 0213:2000 level-4
JA	Unified Japanese IT Vendors Contemporary Ideographs, 1993
JK	Japanese KOKUJI Collection

The 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
K5H	Korean IRG Hanja Character Set 5th Edition: 2001

The Hanja KP sources are

KP0	KPS 9566-97
KP1	KPS 10721:2000 and KPS 10721:2003

The ChuNom V sources are

V0	TCVN 5773:1993
V1	TCVN 6056:1995
V2	VHN 01:1998
V3	VHN 02: 1998
V04	Dictionary on Nom 2006, Dictionary on Nom of Tay ethnic 2006, Lookup Table for Nom in the South 1994

The Unicode U sources are

U0	The Unicode Standard 4.0-2003
UTC	The Unicode Standard 5.1-2008

NOTE 2 – Even if source references get updated, the source reference information is not updated. The updated source references may only identify characters not previously covered by the older version.

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 13-lines header, as many lines as CJK Unified Ideographs in the sum of the two planes; each containing the following information organized in fields delimited by ';' (empty fields use no character):

- 1st field: BMP or SIP code poi~~sitio~~nt `(0hhhh)`, `(2hhhh)`
- 2nd field: Hanzi G sources `(G0-hhhh)`, `(G1-hhhh)`, `(G3-hhhh)`, `(G5-hhhh)`, `(G7-hhhh)`, `(GS-hhhh)`, `(G8-hhhh)`, `(G9-hhhh)`, `(GE-hhhh)`, `(G_KX)`, `(G_KXddddd)`, `(G_HZ)`, `(G_HZddddd)`, `(G_CY)`, `(G_CH)`, `(G_CHddddd)`, `(G_HC)`, `(G_HCddddd)`, `(G_BK)`, `(G_BKddddd)`, `(G_FZ)`, `(G_FZddddd)`, `(G_4K)`, `(G_GHddddd)`, `(G_GJZddddd)`, `(G_XCddddd)`, `(G_CYYddddd)`, `(G_ZFYddddd)`, or `(G_ZJWddddd)`.
- 3rd field: Hanzi T sources `T1-hhhh)`, `(T2-hhhh)`, `(T3-hhhh)`, `(T4-hhhh)`, `(T5-hhhh)`, `(T6-hhhh)`, `(T7-hhhh)`, `(TC-hhhh)`, `(TD-hhhh)`, `(TE-hhhh)`, or `(TF-hhhh)`
- 4th field: Kanji J sources `(J0-hhhh)`, `(J1-hhhh)`, `(J3-hhhh)`, `(J3A-hhhh)`, `(J4-hhhh)`, `(JA-hhhh)` or `(JK-ddddd)`.
- 5th field: Hanja K sources `(K0-hhhh)`, `K1-hhhh)`, `(K2-hhhh)`, `(K3-hhhh)`, `(K4-hhhh)`, or `(K5Hddddd)`.
- 6th field: ChuNom V sources `(V0-hhhh)`, `V1-hhhh)`, `(V2-hhhh)`, `(V3-hhhh)`, or `(V04-hhhh)`.
- 7th field: Hanzi H source `(H-hhhh)`.
- 8th field: Hanja KP sources `(KP0-hhhh)` or `(KP1-hhhh)`.
- 9th field: Unicode U sources `(U0-hhhh)` or `(UTCddddd)`.
- 10th field: Hanzi M source `(MACddddd)`.

The format definition uses `'d'` as a decimal unit and `'h'` as a hexadecimal unit. Uppercase characters, digits and all other symbols between parentheses appear as shown.

NOTE 3 – Concerning JIS X 0213:2000 and 2004 sources, level-4 references correspond to the second plane; other level references correspond to the first plane.

NOTE 4 – The original source references in the Hanja K4 source (PKS 5700-3:1998) are described using a single decimal index without ~~row or column~~section or position values. For better consistency with the other sources, those indexes have been

converted into hexadecimal values in the source reference file. Unlike the other hexadecimal values, they do not decompose in ~~row, column~~section, position values.

Click on this highlighted text to access the reference file.

NOTE 5 – The content is also available as a separate viewable file in the same file directory as this document. The file is named: "CJKU_SR.txt".

~~27.2~~23.2 Source reference presentation for BMP CJK Unified Ideographs

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

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

The leftmost column of an entry shows the code ~~position~~ point in ISO/IEC 10646~~, giving the code representation both~~ in ~~decimal (in row/cell format) 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. When non-empty, the second line shows the coded representation in decimal notation which comprises two digits for section number followed by two digits for position number except for the K4 source where it shows the original decimal source as a single 4 digit value. Hanzi H source characters are identified in the G column using the 'H-' prefix. 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~~23.3 Source reference presentation for SIP CJK Unified Ideographs

In the SIP code charts, CJK Unified Ideographs Extension B are arranged in a manner similar to non ideographs and their presentation does not include source reference information. However, CJK Unified Ideographs Extension C uses a different format:

Ucode	C G	C M	C T	J	K	U	V
2AB7C	𪮷 G_ZFY00619		𪮷 TC-3248				𪮷 V04-4876

The leftmost column of any entry shows the code ~~position~~ point in ISO/IEC 10646. Each of the other columns shows the graphic symbol for the character and its coded representation in the source standard also identified in the table entry.

~~27.4~~23.4 Source references for CJK Compatibility Ideographs

The following list identifies all sources referenced by the CJK Compatibility Ideographs in both the BMP and the SIP. The current full set of CJK Compatibility Ideographs is represented by the collection 383 CJK COMPATIBILITY IDEOGRAPHS-2005 (See A.1~~A.1~~).

The Hanzi H source is

H	Hong Kong Supplementary Character Set - 2004

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 0213:2000 level-3
J4	JIS X 0213:2000 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 fields delimited by ';' (empty fields use no character):

- 1st field: BMP or SIP code ~~position~~ point (0hhhh) or (2hhhh).
- 2nd field: Code ~~position~~ point of corresponding CJK Unified Ideograph (0hhhh) or (2hhhh).
- 3rd field: Hanzi T sources (T3-hhhh), (T4-hhhh), (T5-hhhh), (T6-hhhh), (T7-hhhh), or (TF-hhhh).
- 4th field: Hanzi H source (H-hhhh).
- 5th field: Kanji J sources (J3-hhhh), J4-hhhh).
- 6th field: Hanja K source (K0-hhhh)
- 7th field: Unicode U source (U0-hhhh)
- 8th field: Hanja KP source (KP1-hhhh)

The format definition uses 'h' as a hexadecimal unit. Uppercase characters, digits and all other symbols between parentheses appear as shown.

NOTE 1 – Concerning JIS X 0213:2000 and 2004 sources, level-4 references correspond to the second plane; other level references correspond to the first plane.

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: "CJKC_SR.txt".

~~28~~24 Character names and annotations

~~28.1~~24.1 Entity names

This standard specifies names for the following entity types

- characters
- named UCS sequences identifiers (see 25~~29~~)
- blocks (see 14~~18~~ and A.2~~A.2~~)
- collections (see A.1~~A.1~~)

The names given by this standard to these entities shall follow the rules for name formation and name uniqueness specified in this clause. This specification applies to the entity names in the English language version of this standard.

NOTE 1 – In a version of such a standard in another language a) these rules may be amended to permit names to be generated using words and syntax that are considered appropriate within that language; b) the entity names from this version of the standard may be replaced by equivalent unique names constructed according to the rules amended as in a) above.

NOTE 2 – Additional guidelines for constructing entity names are given in annex L for information.

~~28.2~~24.2 Name formation

An entity names shall consist only of the following characters

- LATIN CAPITAL LETTER A through LATIN CAPITAL LETTER Z,
- DIGIT ZERO through DIGIT NINE,
- SPACE,
- HYPHEN-MINUS, and
- FULL STOP if the entity being named is a collection

The first character in an entity name shall be a Latin capital letter. The last character in an entity name shall be either a Latin capital letter or a Digit.

An entity name shall not contain two or more consecutive SPACE characters or consecutive HYPHEN-MINUS characters. A collection name shall not contain two or more consecutive FULL STOP characters.

A sequence of a SPACE followed by a HYPHEN-MINUS or a sequence of a HYPHEN-MINUS followed by a SPACE may appear only in character names or named UCS sequence identifiers.

EXAMPLE 1 Each of the following two character names contains a consecutive SPACE and HYPHEN-MINUS:

TIBETAN LETTER -A

TIBETAN MARK BKA- SHOG YIG MGO

FULL STOP may appear only in between two alpha-numeric characters (LATIN CAPITAL LETTER A through LATIN CAPITAL LETTER Z, DIGIT ZERO through DIGIT NINE) in a collection name.

EXAMPLE 2 The following collection name contains FULL STOP in between two Digits, DIGIT FOUR and DIGIT ONE:

UNICODE 4.1

EXAMPLE 3 The following collection name contains FULL STOP in between one Latin letter, LATIN CAPITAL LETTER D, and a Digit, DIGIT SEVEN:

BMP-AMD.7

~~28.3~~24.3 Single name

Each entity named in this standard shall be given only one name.

NOTE – This does not preclude the informative use of name aliases or acronyms for the sake of clarity. However, the normative entity name will be unique.

~~28.4~~24.4 Name uniqueness

Each entity name must also be unique within an appropriate name space, as specified here.

~~28.4.1~~24.4.1 Block names

Block names constitute a name space. Each block name shall be unique and distinct from all other block names specified in the standard.

~~28.4.2~~24.4.2 Collection names

Collection names constitute a name space. Each collection name shall be unique and distinct from all other collection names specified in the standard.

~~28.4.3~~24.4.3 Character names and named UCS sequence identifiers

Character names and named UCS sequence identifiers, taken together, constitute a name space. Each character name or named UCS sequence identifier shall be unique and distinct from all other character names or named UCS sequence identifiers.

~~28.4.4~~24.4.4 Determining uniqueness

For block names and collection names, two names shall be considered unique and distinct if they are different even when SPACE and medial HYPHEN-MINUS characters are ignored in comparison of the names.

NOTE 1 – A medial HYPHEN-MINUS is a HYPHEN-MINUS character that occurs immediately after a character other than SPACE and immediately before a character other than SPACE.

EXAMPLE 1 The following hypothetical block names would be unique and distinct:

LATIN-A
LATIN-B

EXAMPLE 2 The following hypothetical block names would not be unique and distinct:

LATIN-A
LATIN A
LATINA

For character names and named UCS sequence identifiers, two names shall be considered unique and distinct if they are different even when SPACE and medial HYPHEN-MINUS characters are ignored and even when the words "LETTER", "CHARACTER", and "DIGIT" are ignored in comparison of the names.

EXAMPLE 3 The following hypothetical character names would not be unique and distinct:

MANICHAEAN CHARACTER A
MANICHAEAN LETTER A

EXAMPLE 4: The following two actual character names are unique and distinct, because they differ by a HYPHEN-MINUS that is not a medial HYPHEN-MINUS:

TIBETAN LETTER A
TIBETAN LETTER -A

The following two character names shall be considered unique and distinct:

HANGUL JUNGSEONG OE
HANGUL JUNGSEONG O-E

NOTE 2 – These two character names are explicitly handled as an exception, because they were defined in an earlier version of this International Standard before the introduction of the name uniqueness requirement. This pair is, has been, and will be the only exception to the uniqueness rule in this International Standard.

~~28.5~~24.5 Annotations

A character name or a named UCS sequence identifier may be followed by an additional explanatory statement not part of the name, and separated by a single SPACE character. These statements are in parentheses and use the Latin lower case letters a-z, digits 0-9, SPACE and HYPHEN-MINUS. A capital Latin letter A-Z may be used for word initials where required.

Such parenthetical annotations are not part of the entity names themselves, and the characters used in the annotations are not subject to the name uniqueness requirements.

~~A character name may also be followed by a single ASTERISK separated from the name by a single SPACE. If a parenthetical annotation is present, the ASTERISK follows the annotation and is separated from the closing parenthesis by a single SPACE.~~

~~The presence of the ASTERISK notes that additional information on the character is available in annex P of this standard.~~

~~28.6~~24.6 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 expressed as four hexadecimal digits. 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 hexadecimal digit value. For example, the first CJK Ideograph character in the SIP has the name “CJK UNIFIED IDEOGRAPH-20000”.

~~28.7~~24.7 Character names and annotations for Hangul syllables

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

1) Obtain the code ~~position number~~point 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 representing the ~~; h_1h_2 is the Row number within the BMP and h_3h_4 is the cell number within the row. The~~ number $h_1h_2h_3h_4$ lying~~ies~~ 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
$C = 4096 \times (d_1 - 10) + 256 \times (d_2 - 12) + 16 \times d_3 + d_4$

4) Calculate the syllable component indices I, P, F from the following formulae
$I = C / 588$ (Note: $0 \le I \le 18$)
$P = (C \% 588) / 28$ (Note: $0 \le P \le 20$)
$F = C \% 28$ (Note: $0 \le F \le 27$)
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~~code point ~~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~~ with code ~~position~~ point 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 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~~ points ~~0000~~ AC00 - ~~0000~~ D7A3 are also derived from their code ~~position~~ point values~~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~~ with code ~~position~~ point 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).

NOTE – The annex R provides the names of Hangul syllables in two formats: syllable-name and full name/annotation, both available through linked files.

Table ~~1~~5: Elements of Hangul syllable names and annotations

Index number	Syllable name elements			Annotation elements		
	I string	*P* string	*F* string	*I* string	*P* string	*F* 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~~25 Named UCS Sequence Identifiers

A Named UCS Sequence Identifier (NUSI) is a USI associated to a name following the same construction rules as for character names. These rules are given in 24~~28~~.

NOTE – The purpose of these named USIs is to specify sequences of characters that may be treated as single units, either in particular types of processing, in reference by standards, in listing of repertoires (such as for fonts or keyboards).

The USI value corresponding to each NUSI is written using the coded representation determined by the normalization form NFC (see 21~~25~~). Each named UCS sequence has a unique code representation. All the allowed named UCS sequence identifiers are shown in this clause; all other such named sequences are undefined. The following list provides a description of these named UCS sequence identifiers.

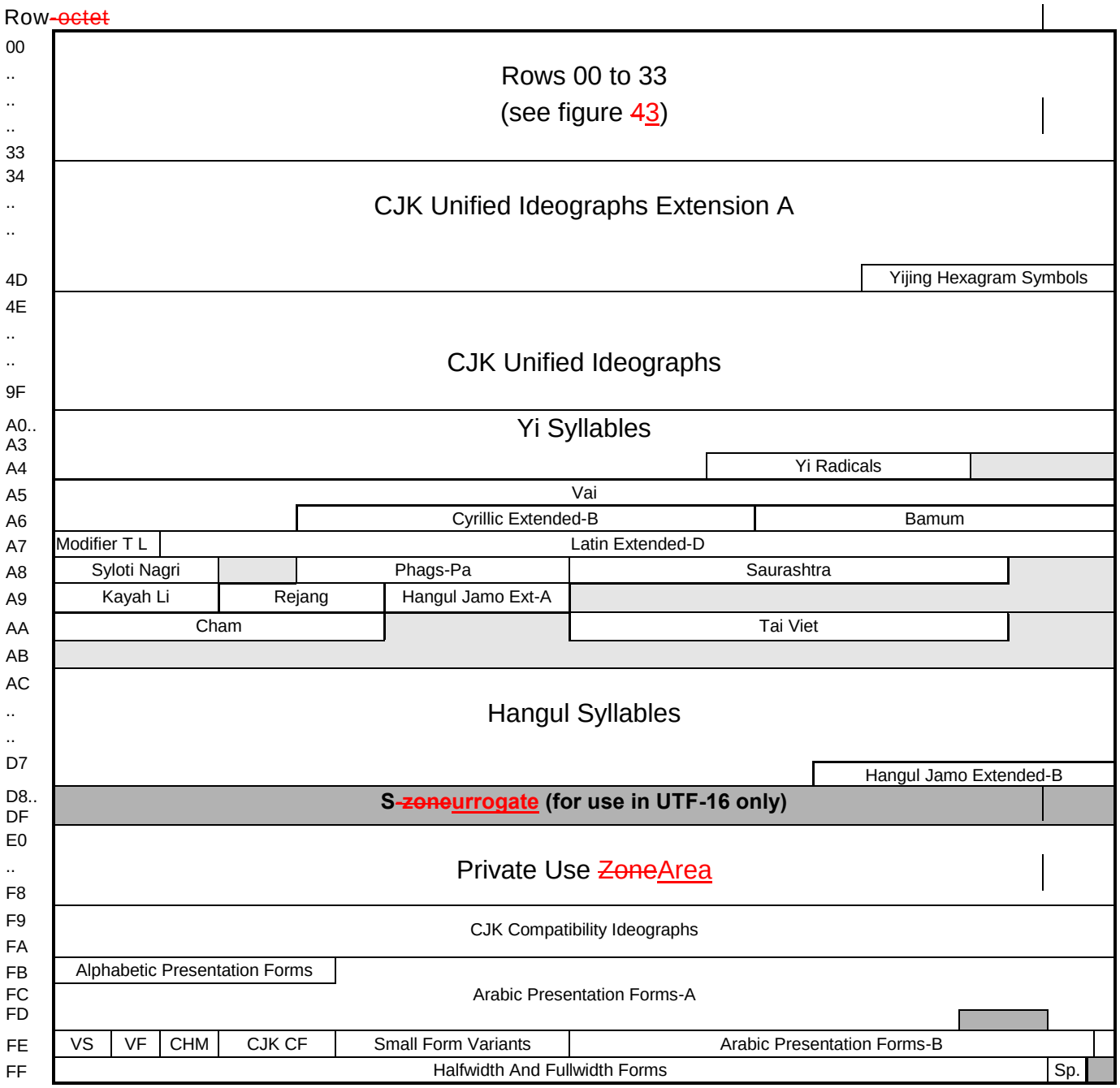
USI	USI name
<0100, 0300>	LATIN CAPITAL LETTER A WITH MACRON AND GRAVE
<0101, 0300>	LATIN SMALL LETTER A WITH MACRON AND GRAVE
<0104, 0301>	LATIN CAPITAL LETTER A WITH OGONEK AND ACUTE
<0105, 0301>	LATIN SMALL LETTER A WITH OGONEK AND ACUTE
<0104, 0303>	LATIN CAPITAL LETTER A WITH OGONEK AND TILDE
<0105, 0303>	LATIN SMALL LETTER A WITH OGONEK AND TILDE
<0045, 0329>	LATIN CAPITAL LETTER E WITH VERTICAL LINE BELOW
<0065, 0329>	LATIN SMALL LETTER E WITH VERTICAL LINE BELOW
<00C8, 0329>	LATIN CAPITAL LETTER E WITH VERTICAL LINE BELOW AND GRAVE

<00E8, 0329> LATIN SMALL LETTER E WITH VERTICAL LINE BELOW AND GRAVE
<00C9, 0329> LATIN CAPITAL LETTER E WITH VERTICAL LINE BELOW AND ACUTE
<00E9, 0329> LATIN SMALL LETTER E WITH VERTICAL LINE BELOW AND ACUTE
<00CA, 0304> LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND MACRON
<00EA, 0304> LATIN SMALL LETTER E WITH CIRCUMFLEX AND MACRON
<00CA, 030C> LATIN CAPITAL LETTER E WITH CIRCUMFLEX AND CARON
<00EA, 030C> LATIN SMALL LETTER E WITH CIRCUMFLEX AND CARON
<0118, 0301> LATIN CAPITAL LETTER E WITH OGONEK AND ACUTE
<0119, 0301> LATIN SMALL LETTER E WITH OGONEK AND ACUTE
<0118, 0303> LATIN CAPITAL LETTER E WITH OGONEK AND TILDE
<0119, 0303> LATIN SMALL LETTER E WITH OGONEK AND TILDE
<0116, 0301> LATIN CAPITAL LETTER E WITH DOT ABOVE AND ACUTE
<0117, 0301> LATIN SMALL LETTER E WITH DOT ABOVE AND ACUTE
<0116, 0303> LATIN CAPITAL LETTER E WITH DOT ABOVE AND TILDE
<0117, 0303> LATIN SMALL LETTER E WITH DOT ABOVE AND TILDE
<012A, 0300> LATIN CAPITAL LETTER I WITH MACRON AND GRAVE
<012B, 0300> LATIN SMALL LETTER I WITH MACRON AND GRAVE
<0069, 0307, 0301> LATIN SMALL LETTER I WITH DOT ABOVE AND ACUTE
<0069, 0307, 0300> LATIN SMALL LETTER I WITH DOT ABOVE AND GRAVE
<0069, 0307, 0303> LATIN SMALL LETTER I WITH DOT ABOVE AND TILDE
<012E, 0301> LATIN CAPITAL LETTER I WITH OGONEK AND ACUTE
<012F, 0307, 0301> LATIN SMALL LETTER I WITH OGONEK AND DOT ABOVE AND ACUTE
<012E, 0303> LATIN CAPITAL LETTER I WITH OGONEK AND TILDE
<012F, 0307, 0303> LATIN SMALL LETTER I WITH OGONEK AND DOT ABOVE AND TILDE
<004A, 0303> LATIN CAPITAL LETTER J WITH TILDE
<006A, 0307, 0303> LATIN SMALL LETTER J WITH DOT ABOVE AND TILDE
<004C, 0303> LATIN CAPITAL LETTER L WITH TILDE
<006C, 0303> LATIN SMALL LETTER L WITH TILDE
<004D, 0303> LATIN CAPITAL LETTER M WITH TILDE
<006D, 0303> LATIN SMALL LETTER M WITH TILDE
<006E, 0360, 0067> LATIN SMALL LETTER NG WITH TILDE ABOVE
<004F, 0329> LATIN CAPITAL LETTER O WITH VERTICAL LINE BELOW
<006F, 0329> LATIN SMALL LETTER O WITH VERTICAL LINE BELOW
<00D2, 0329> LATIN CAPITAL LETTER O WITH VERTICAL LINE BELOW AND GRAVE
<00F2, 0329> LATIN SMALL LETTER O WITH VERTICAL LINE BELOW AND GRAVE
<00D3, 0329> LATIN CAPITAL LETTER O WITH VERTICAL LINE BELOW AND ACUTE
<00F3, 0329> LATIN SMALL LETTER O WITH VERTICAL LINE BELOW AND ACUTE
<0052, 0303> LATIN CAPITAL LETTER R WITH TILDE
<0072, 0303> LATIN SMALL LETTER R WITH TILDE
<0053, 0329> LATIN CAPITAL LETTER S WITH VERTICAL LINE BELOW
<0073, 0329> LATIN SMALL LETTER S WITH VERTICAL LINE BELOW
<016A, 0300> LATIN CAPITAL LETTER U WITH MACRON AND GRAVE
<016B, 0300> LATIN SMALL LETTER U WITH MACRON AND GRAVE
<0172, 0301> LATIN CAPITAL LETTER U WITH OGONEK AND ACUTE
<0173, 0301> LATIN SMALL LETTER U WITH OGONEK AND ACUTE
<0172, 0303> LATIN CAPITAL LETTER U WITH OGONEK AND TILDE
<0173, 0303> LATIN SMALL LETTER U WITH OGONEK AND TILDE
<016A, 0301> LATIN CAPITAL LETTER U WITH MACRON AND ACUTE
<016B, 0301> LATIN SMALL LETTER U WITH MACRON AND ACUTE
<016A, 0303> LATIN CAPITAL LETTER U WITH MACRON AND TILDE
<016B, 0303> LATIN SMALL LETTER U WITH MACRON AND TILDE
<10E3, 0302> GEORGIAN LETTER U-BRJGU
<17D2, 1780> KHMER CONSONANT SIGN COENG KA
<17D2, 1781> KHMER CONSONANT SIGN COENG KHA
<17D2, 1782> KHMER CONSONANT SIGN COENG KO
<17D2, 1783> KHMER CONSONANT SIGN COENG KHO
<17D2, 1784> KHMER CONSONANT SIGN COENG NGO
<17D2, 1785> KHMER CONSONANT SIGN COENG CA
<17D2, 1786> KHMER CONSONANT SIGN COENG CHA

<17D2, 1787>	KHMER CONSONANT SIGN COENG CO
<17D2, 1788>	KHMER CONSONANT SIGN COENG CHO
<17D2, 1789>	KHMER CONSONANT SIGN COENG NYO
<17D2, 178A>	KHMER CONSONANT SIGN COENG DA
<17D2, 178B>	KHMER CONSONANT SIGN COENG TTHA
<17D2, 178C>	KHMER CONSONANT SIGN COENG DO
<17D2, 178D>	KHMER CONSONANT SIGN COENG TTHO
<17D2, 178E>	KHMER CONSONANT SIGN COENG NA
<17D2, 178F>	KHMER CONSONANT SIGN COENG TA
<17D2, 1790>	KHMER CONSONANT SIGN COENG THA
<17D2, 1791>	KHMER CONSONANT SIGN COENG TO
<17D2, 1792>	KHMER CONSONANT SIGN COENG THO
<17D2, 1793>	KHMER CONSONANT SIGN COENG NO
<17D2, 1794>	KHMER CONSONANT SIGN COENG BA
<17D2, 1795>	KHMER CONSONANT SIGN COENG PHA
<17D2, 1796>	KHMER CONSONANT SIGN COENG PO
<17D2, 1797>	KHMER CONSONANT SIGN COENG PHO
<17D2, 1798>	KHMER CONSONANT SIGN COENG MO
<17D2, 1799>	KHMER CONSONANT SIGN COENG YO
<17D2, 179A>	KHMER CONSONANT SIGN COENG RO
<17D2, 179B>	KHMER CONSONANT SIGN COENG LO
<17D2, 179C>	KHMER CONSONANT SIGN COENG VO
<17D2, 179D>	KHMER CONSONANT SIGN COENG SHA
<17D2, 179E>	KHMER CONSONANT SIGN COENG SSA
<17D2, 179F>	KHMER CONSONANT SIGN COENG SA
<17D2, 17A0>	KHMER CONSONANT SIGN COENG HA
<17D2, 17A1>	KHMER CONSONANT SIGN COENG LA
<17D2, 17A2>	KHMER VOWEL SIGN COENG QA
<17D2, 17A7>	KHMER INDEPENDENT VOWEL SIGN COENG QU
<17D2, 17AB>	KHMER INDEPENDENT VOWEL SIGN COENG RY
<17D2, 17AC>	KHMER INDEPENDENT VOWEL SIGN COENG RYY
<17D2, 17AF>	KHMER INDEPENDENT VOWEL SIGN COENG QE
<17BB 17C6>	KHMER VOWEL SIGN OM
<17B6, 17C6>	KHMER VOWEL SIGN AAM
<31F7, 309A>	KATAKANA LETTER AINU P
<02E5, 02E9>	MODIFIER LETTER EXTRA-HIGH EXTRA-LOW CONTOUR TONE BAR

~~30~~26 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.



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

Figure ~~3~~ 2 - Overview of the Basic Multilingual Plane

Row ~~octet~~																
00	Controls	Basic Latin							Controls	Latin-1 Supplement						
01	Latin Extended-A								Latin Extended-B							
02	Latin Extended-B				IPA (Intl. Phonetic Alphabet) Extensions					Spacing Modifier Letters						
03	Combining Diacritical Marks							Greek and Coptic								
04	Cyrillic															
05	Cyrillic Supplement			Armenian						Hebrew						
06	Arabic															
07	Syriac				Arabic Sup.				Thaana				Nko			
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									Ethiopic Sup.		Cherokee					
14..	Unified Canadian Aboriginal Syllabics															
16									Ogham		Runic					
17	Tagalog	Hanunoo		Buhid		Tagbanwa			Khmer							
18	Mongolian															
19	Limbu				Tai Le				New Tai Lue *					Khmer Symb.		
1A	Buginese	Lanna														
1B	Balinese								Sundanese							
1C	Lepcha				Ol Chiki				Meitei Mayek							
1D	Phonetic Extension								Phonetic Extensions Sup.				Combining Diacritical M Sup.			
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. Symbols-A			S A A
28	Braille Patterns															
29	Supplemental Arrows-B								Miscellaneous Mathematical Symbols-B							
2A	Supplemental Mathematical Operators															
2B	Miscellaneous Symbols and Arrows															
2C	Glagolitic					Latin Ext-C			Coptic							
2D	Georgian Sup.			Tifinagh					Ethiopic Extended						Cyrillic Ext-A	
2E	Supplemental Punctuation								CJK Radicals Supplement							
2F	Kangxi Radicals													Ideog. Descr.		
30	CJK Symbols And Punctuation				Hiragana						Katakana					
31	Bopomofo			Hangul Compatibility Jamo				Kanbun			Bopomofo E.		CJK Strokes			K P E
32	Enclosed CJK Letters And Months															
33	CJK Compatibility															

= reserved for future standardization

* NOTE 1 – New Tai Lue is also known as Xishuang Banna Dai
NOTE 2 – Vertical boundaries within rows are indicated in approximate positions only.

Figure ~~4~~ 3 - Overview of Rows 00 to 33 of the Basic Multilingual Plane

~~31~~27 Structure of the Supplementary Multilingual Plane for s~~S~~cripts and symbols (SMP)

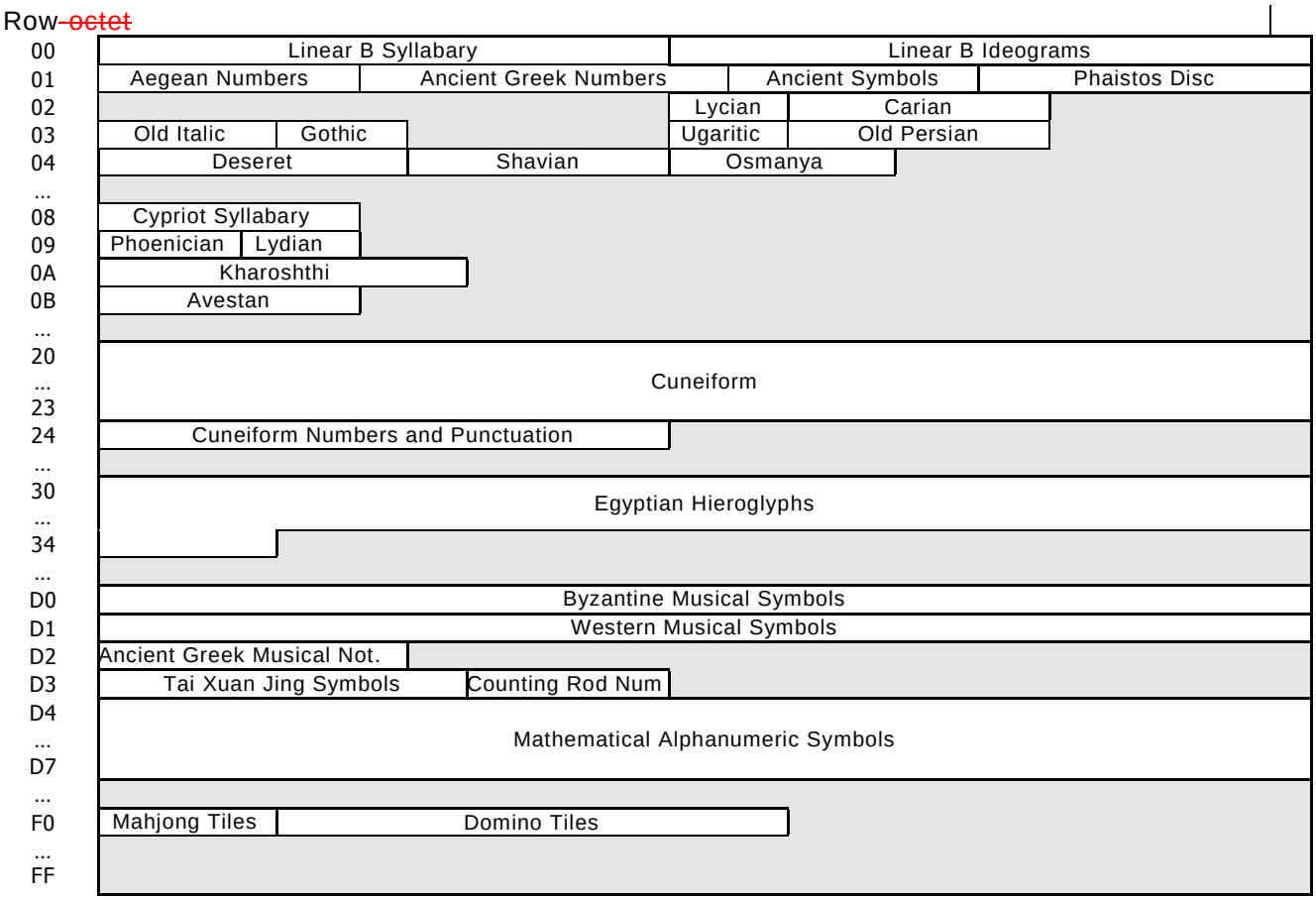
~~The Plane 02 of Group 00 is the Supplementary Multilingual Plane (SMP).~~

Because another supplementary plane is reserved for additional CJK Ideographs, the SMP (plane 1) is not used to date for encoding CJK Ideographs. Instead, the SMP is used for encoding graphic characters used in other scripts of the world that are not encoded in the BMP. Most, but not all, of the scripts encoded to date in the SMP are not in use as living scripts by modern user communities.

NOTE 1 – 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.



NOTE 2 – Vertical boundaries within rows are indicated in approximate positions only.

NOTE 3 – 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

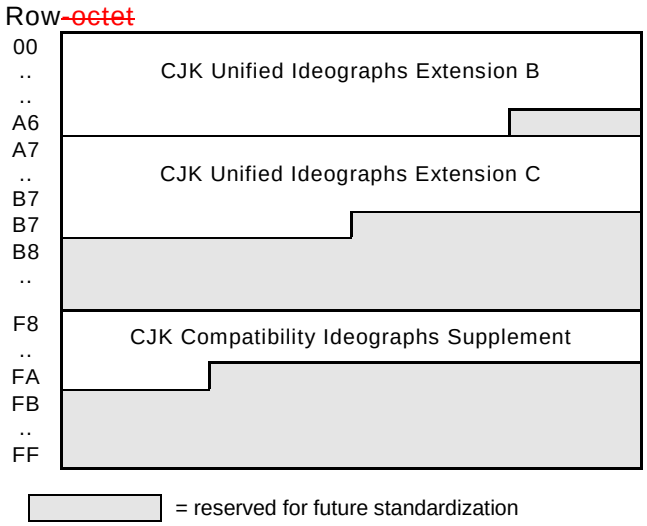
~~32~~28 Structure of the Supplementary Ideographic Plane (SIP)

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

The SIP (plane 2) is used for CJK unified ideographs (unified East Asian ideographs) that are not encoded in the BMP. The procedures for the unification and the rules for their arrangement are described in Annex S~~Annex S~~.

The SIP is also used for compatibility CJK ideographs. These ideographs are compatibility characters as specified in 18~~22~~.

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



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

Figure 6 – Overview of the Supplementary Ideographic Plane

~~33~~29 Structure of the Supplementary Special-purpose Plane (SSP)

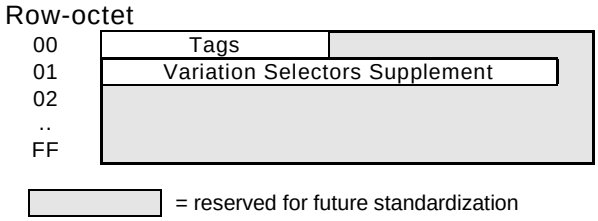
~~The Plane 0E of Group 0 is the Supplementary Special-purpose Plane (SSP).~~

The SSP (plane 0E) is used for special purpose use graphic characters. Code ~~positions~~ points from E0000 to E0FFF are reserved for Format Characters (see 16~~20~~).

NOTE 1 – 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 2 – Unassigned code points in this range should be ignored in normal processing and display.



NOTE 3 – Vertical boundaries within rows are indicated in approximate positions only.

Figure 7 – Overview of the Supplementary Special-purpose Plane

~~34~~30 Code charts and lists of character names

Detailed code charts and lists of character names for the BMP, SMP, SIP and SSP are shown on the following pages. Code charts are arranged by blocks which may span several pages.

Each code chart is followed by a corresponding character names list, except the CJK UNIFIED IDEOGRAPHS blocks and the HANGUL SYLLABLES blocks.

~~34.1~~30.1 Code chart

Code charts are presented in arrays of graphic symbols representing the characters organized in one to sixteen columns of sixteen symbols each. The lower digit of the coded representation is indicated in the left margin while the remaining upper digits are indicated in the top margin. The full coded representation for each character is also indicated under each representative graphic symbol.

~~34.2~~30.2 Character names list

The character names lists contain some normative information such as the code ~~position~~ point and the character name. They also provide additional information clarifying some feature of a character, such as its naming or usage, or its associated graphic symbol. In addition to the code ~~position~~point, the graphic symbol, and the character name, the following informative items may appear in these names list:

- Subheads grouping various subsets of a given block. For example, the LATIN-1 SUPPLEMENT block contain “Latin-1 punctuation and symbols”, “Letters”, and “Mathematical operator”.
- Explanatory text describing context for a subhead or a whole block.
- Aliases, either preceded by ‘=’ or ‘※’ indicate alternate names for characters.
- Cross references, preceded by ‘→’ indicates a related character of interest.
- Information about languages, preceded by ‘•’ indicates a non exhaustive list of languages using that character. For bicameral scripts, the information is only provided for the lower case form of the character.
- Case mappings, also preceded by ‘•’, only when it cannot be derived simply from the names.
- Other information about a character, also preceded by ‘•’, describing name peculiarity, historical consideration, or any noteworthy aspect of a character.
- Decompositions, preceded by ‘≡’, or ‘≈’ describing various mapping between characters.

The following example describes various fragments of name lists including these informative items.

EXAMPLE

Latin-1 punctuation and symbols
Based on ISO/IEC 8859-1 (aka Latin-1) from here.

...

00B5 µ MICRO SIGN
≈ 03BC µ greek small letter mu
00B6 ¶ PILCROW SIGN
= paragraph sign
• section sign in some European usage
→ 204B ⁋ reverse pilcrow sign
→ 2761 ❡ curve stern paragraph sign ornament

…

Letters

…

00DF ß LATIN SMALL LETTER SHARP S
= Eszett
• German

• uppercase is “SS”
• in origin a ligature of 017f ſ and 0073 s
→ 03B2 β greek small letter beta

…

00E5 å LATIN SMALL LETTER A WITH RING ABOVE
• Danish, Norwegian, Swedish, Walloon
≡ 0061 a 030A °

…

01C9 lj LATIN SMALL LETTER IJ
→ 0459 ль cyrillic small letter lje
≈ 006C l 006A j

…

FE18 ︸ PRESENTATION FORM FOR VERTICAL RIGHT WHITE LENTICULAR BRAKCET
※ PRESENTATION FORM FOR VERTICAL RIGHT WHITE LENTICULAR BRACKET
• misspelling of “BRACKET” in character name is a known defect
≈ <vertical> 3017 〗

~~34.3~~30.3 Pointers to code charts and lists of character names

Access to the code charts and lists of character names is provided by clicking on the appropriate highlighted text below.

- Basic Latin to CJK Compatibility (0000-33FF)
- CJK Unified Ideographs Extension A (3400-4DBF)
- Yijing Hexagram Symbols (4DC0-4DFF)
- CJK Unified Ideographs Part 1 of 3 (4E00-680F)
- CJK Unified Ideographs Part 2 of 3 (6810-824F)
- CJK Unified Ideographs Part 3 of 3 (8250-9FFF)
- Yi Syllables to Specials (A000-FFFD)
- Linear B Syllabary to Mathematical Alphanumeric Symbols (10000-1D7FF)
- CJK Unified Ideographs Extension B (20000-2A6DF)
- CJK Compatibility Ideographs (2F800-2FA1F)
- Tag to Variation Selectors Supplement (E0000-E01EF)

NOTE – To preserve the odd-even layout of the code charts, a page from the previous block may be inserted before the actual start of a code chart.

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~~code points" column indicates that the collection is a fixed collection.

Collection number and name		~~Position-~~ ~~s~~Code points
1	BASIC LATIN	0020-007E *
2	LATIN-1 SUPPLEMENT	00A0-00FF *
3	LATIN EXTENDED-A	0100-017F *
4	LATIN EXTENDED-B	0180-024F *
5	IPA EXTENSIONS	0250-02AF *
6	SPACING MODIFIER LETTERS	02B0-02FF *
7	COMBINING DIACRITICAL MARKS	0300-036F *
8	BASIC GREEK	0370-03CF
9	GREEK SYMBOLS AND COPTIC	03D0-03FF
10	CYRILLIC	0400-04FF *
11	ARMENIAN	0530-058F
12	BASIC HEBREW	05D0-05EA *
13	HEBREW EXTENDED	0590-05CF 05EB-05FF
14	BASIC ARABIC	0600-065F
15	ARABIC EXTENDED	0660-06FF *
16	DEVANAGARI	0900-097F 200C, 200D
17	BENGALI	0980-09FF 200C, 200D
18	GURMUKHI	0A00-0A7F 200C, 200D
19	GUJARATI	0A80-0AFF 200C, 200D
20	ORIYA	0B00-0B7F 200C, 200D
21	TAMIL	0B80-0BFF 200C, 200D
22	TELUGU	0C00-0C7F 200C, 200D
23	KANNADA	0C80-0CFF 200C, 200D
24	MALAYALAM	0D00-0D7F 200C, 200D
25	THAI	0E00-0E7F
26	LAO	0E80-0EFF
27	BASIC GEORGIAN	10D0-10FF
28	GEORGIAN EXTENDED	10A0-10CF
29	HANGUL JAMO	1100-11FF *
30	LATIN EXTENDED ADDITIONAL	1E00-1EFF *
31	GREEK EXTENDED	1F00-1FFF
32	GENERAL PUNCTUATION	2000-206F
33	SUPERSCRIPTS AND SUBSCRIPTS	2070-209F
34	CURRENCY SYMBOLS	20A0-20CF
35	COMBINING DIACRITICAL MARKS FOR SYMBOLS	20D0-20FF
36	LETTERLIKE SYMBOLS	2100-214F *
37	NUMBER FORMS	2150-218F
38	ARROWS	2190-21FF *
39	MATHEMATICAL OPERATORS	2200-22FF *
40	MISCELLANEOUS TECHNICAL	2300-23FF
41	CONTROL PICTURES	2400-243F
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
49	CJK SYMBOLS AND PUNCTUATION	3000-303F *
50	HIRAGANA	3040-309F
51	KATAKANA	30A0-30FF *
52	BOPOMOFO	3100-312F 31A0-31BF
53	HANGUL COMPATIBILITY JAMO	3130-318F
54	CJK MISCELLANEOUS	3190-319F
55	ENCLOSED CJK LETTERS AND MONTHS	3200-32FF
56	CJK COMPATIBILITY	3300-33FF *
57, 58, 59 (These collection numbers shall not be used, see Note 2.)		
60	CJK UNIFIED IDEOGRAPHS	4E00-9FFF
61	PRIVATE USE AREA	E000-F8FF
62	CJK COMPATIBILITY IDEOGRAPHS	F900-FAFF
63	(Collection specified as union of other collections)	
64	ARABIC PRESENTATION FORMS-A	FB50-FDCF FDF0-FDFF
65	COMBINING HALF MARKS	FE20-FE2F
66	CJK COMPATIBILITY FORMS	FE30-FE4F *
67	SMALL FORM VARIANTS	FE50-FE6F
68	ARABIC PRESENTATION FORMS-B	FE70-FEFE
69	HALFWIDTH AND FULLWIDTH FORMS	FF00-FFEF
70	SPECIALS	FFF0-FFFD
71	HANGUL SYLLABLES	AC00-D7A3 *
72	BASIC TIBETAN	0F00-0FBF

73	ETHIOPIC	1200-137F
74	UNIFIED CANADIAN ABORIGINAL SYLLABICS	1400-167F
75	CHEROKEE	13A0-13FF
76	YI SYLLABLES	A000-A48F
77	YI RADICALS	A490-A4CF
78	KANGXI RADICALS	2F00-2FDF
79	CJK RADICALS SUPPLEMENT	2E80-2EFF
80	BRAILLE PATTERNS	2800-28FF
81	CJK UNIFIED IDEOGRAPHS EXTENSION A	3400-4DBF FA1F, FA23
82	OGHAM	1680-169F
83	RUNIC	16A0-16FF
84	SINHALA	0D80-0DFF
85	SYRIAC	0700-074F
86	THAANA	0780-07BF
87	BASIC MYANMAR	1000-104F 200C, 200D
88	KHMER	1780-17FF 200C, 200D
89	MONGOLIAN	1800-18AF
90	EXTENDED MYANMAR	1050-109F
91	TIBETAN	0F00-0FFF
92	CYRILLIC SUPPLEMENT	0500-052F
93	TAGALOG	1700-171F
94	HANUNOO	1720-173F
95	BUHID	1740-175F
96	TAGBANWA	1760-177F
97	MISCELLANEOUS MATHEMATICAL SYMBOLS-A	27C0-27EF
98	SUPPLEMENTAL ARROWS-A	27F0-27FF *
99	SUPPLEMENTAL ARROWS-B	2900-297F *
100	MISCELLANEOUS MATHEMATICAL SYMBOLS-B	2980-29FF *
101	SUPPLEMENTAL MATHEMATICAL OPERATORS	2A00-2AFF *
102	KATAKANA PHONETIC EXTENSIONS	31F0-31FF *
103	VARIATION SELECTORS	FE00-FE0F *
104	LTR ALPHABETIC PRESENTATION FORMS	FB00-FB1C
105	RTL ALPHABETIC PRESENTATION FORMS	FB1D–FB4F
106	LIMBU	1900-194F
107	TAI LE	1950-197F
108	KHMER SYMBOLS	19E0-19FF *
109	PHONETIC EXTENSIONS	1D00-1D7F *
110	MISCELLANEOUS SYMBOLS AND ARROWS	2B00-2BFF
111	YIJING HEXAGRAM SYMBOLS	4DC0-4DFF *
112	ARABIC SUPPLEMENT	0750-077F *
113	ETHIOPIC SUPPLEMENT	1380-139F
114	NEW TAI LUE	1980-19DF
115	BUGINESE	1A00-1A1F
116	PHONETIC EXTENSIONS SUPPLEMENT *	1D80-1DBF
117	COMBINING DIACRITICAL MARKS SUPPLEMENT	1DC0-1DFF
118	GLAGOLITIC	2C00-2C5F
119	COPTIC	03E2-03EF 2C80-2CFF
120	GEORGIAN SUPPLEMENT	2D00-2D2F
121	TIFINAGH	2D30-2D7F
122	ETHIOPIC EXTENDED	2D80-2DDF
123	SUPPLEMENTAL PUNCTUATION	2E00-2E7F
124	CJK STROKES	31C0-31EF
125	MODIFIER TONE LETTERS	A700-A71F *
126	SYLOTI NAGRI	A800-A82F
127	VERTICAL FORMS	FE10-FE1F
128	NKO	07C0-07FF
129	BALINESE	1B00-1B7F
130	LATIN EXTENDED-C	2C60-2C7F
131	LATIN EXTENDED-D	A720-A7FF
132	PHAGS-PA	A840-A87F
133	SUNDANESE	1B80-1BBF
134	LEPCHA	1C00-1C4F
135	OL CHIKI	1C50-1C7F *
136	VAI	A500-A63F
137	SAURASHTRA	A880-A8DF
138	KAYAH LI	A900-A92F *
139	REJANG	A930-A95F
140	LANNA	1A20-1AAF
141	CYRILLIC EXTENDED-A	2DE0-2DFF *
142	CYRILLIC EXTENDED-B	A640-A69F
143	CHAM	AA00-AA5F
144	MEITEI MAYEK	1C80-1CCF
145	BAMUM	A6A0-A6FF
146	HANGUL JAMO EXTENDED-A	A960-A97F
147	TAI VIET	AA80-AADF
148	HANGUL JAMO EXTENDED-B	D7B0-D7FF
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

1015	ANCIENT GREEK NUMBERS	10140-1018F
1016	OLD PERSIAN	103A0-103DF
1017	KHAROSHTHI	10A00-10A5F
1018	ANCIENT GREEK MUSICAL NOTATION	1D200-1D24F
1019	PHOENICIAN	10900-1091F
1020	CUNEIFORM	12000-123FF
1021	CUNEIFORM NUMBERS AND PUNCTUATION	12400-1247F
1022	COUNTING ROD NUMERALS	1D360-1D37F
1023	PHAISTOS DISC	101D0-101FF
1024	LYCIAN	10280-1029F
1025	CARIAN	102A0-102DF
1026	LYDIAN	10920-1093F
1027	ANCIENT SYMBOLS	10190-101CF
1028	MAHJONG TILES	1F000-1F02F
1029	DOMINO TILES	1F030-1F09F
1030	AVESTAN	10B00-10B3F
1031	EGYPTIAN HIEROGLYPHS	13000-1342F
2001	CJK UNIFIED IDEOGRAPHS EXTENSION B	20000-2A6DF
2002	CJK COMPATIBILITY IDEOGRAPHS SUPPLEMENT	2F800-2FA1F
2003	CJK UNIFIED IDEOGRAPHS EXTENSION C	2A700-2B77F
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
208	CONTROL CHARACTERS	0000-001F 0007F-009F
3002	ALTERNATE FORMAT CHARACTERS	E0000-E0FFF

The following specify collections that represented the whole UCS when they were created

299	(This collection number shall not be used, see A.1.1~~A.3.2~~.)	
301	BMP-AMD.7	see A.3.1~~A.3.1~~ *
302	BMP SECOND EDITION	see A.3.3~~A.3.3~~ *
303	UNICODE 3.1	see A.6.1~~A.6.1~~ *
304	UNICODE 3.2	see A.6.2~~A.6.2~~ *
305	UNICODE 4.0	see A.1.1~~A.6.3~~ *
306	UNICODE 4.1	see A.6.4~~A.6.4~~ *
307	UNICODE 5.0	see A.1.1~~A.6.5~~ *
308	UNICODE 5.1	see A.1.1~~A.6.6~~ *
340	COMBINED FIRST EDITION	see A.1.1~~A.3.4~~ *
10646	UNICODE	0000-FDCF FDF0-FFFD 10000-1FFFD 20000-2FFFD 30000-3FFFD 40000-4FFFD 50000-5FFFD 60000-6FFFD 70000-7FFFD 80000-8FFFD 90000-9FFFD A0000-AFFFD B0000-BFFFD C0000-CFFFD D0000-DFFFD E0000-EFFFD F0000-FFFFD 100000-10FFFD

NOTE 1 – The UNICODE collection incorporates all characters currently encoded in the standard

The following collections only contain CJK ideographs.

370	IICORE	see A.4.1~~A.4.1~~ *
371	JIS2004 IDEOGRAPHICS EXTENSION	see A.4.2~~A.4.2~~ *
372	JAPANESE IDEOGRAPHICS SUPPLEMENT	see A.4.3~~A.4.3~~ *
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
382	CJK UNIFIED IDEOGRAPHS-2005	Collection 380* 9FA6-9FBB
383	CJK COMPATIBILITY IDEOGRAPHS-2005	Collection 381 * FA70-FAD9
384	CJK UNIFIED IDEOGRAPHS-2007	Collection 382 * 9FBC-9FC3
385	CJK UNIFIED IDEOGRAPHS-2008	Collection 384 * 2A700-2B77A

The following specify other collections, including extended collections.

270	COMBINING CHARACTERS	BMP characters specified in clause 4.14 ~~Annex B~~
271	(This collection number shall not be used, see Note 2)	
281	MES-1	see A.5.1~~A.5.1~~ *
282	MES-2	see A.5.2~~A.5.2~~ *
283	MODERN EUROPEAN SCRIPTS	see A.5.3~~A.5.3~~ *
284	CONTEMPORARY LITHUANIAN LETTERS	see A.1.1~~A.5.4~~ *
285	BASIC JAPANESE	see A.5.5~~A.5.5~~ *
286	JAPANESE NON IDEOGRAPHICS EXTENSION	see A.1.1~~A.5.6~~ *
287	COMMON JAPANESE	see A.1.1~~A.5.7~~ *
300	BMP	0000-D7FF E000-FFFD
400	(This collection number shall not be used, see Note 3.)	
401	PRIVATE USE PLANES-0F-10	G=00, P=0F-10
500	(This collection number shall not be used, see Note 3.)	
1000	SMP	10000-1FFFD
1900	SMP COMBINING CHARACTERS	~~-~~SMP characters specified in clause 4.14 ~~Annex B~~
2000	SIP	20000-2FFFD
3000	SSP	E0000-EFFFD

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

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. The collection numbered 271 was specified in the first edition of ISO/IEC 10646 but has 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	Bopomofo	52
Alphabetic	63	Braille patterns	80
Alphanumeric	43	Buginese	115
Ancient Greek	1015 1018	Buhid	95
Arabic	14 15 64 68 112	Byzantine musical symbols	1004
Armenian	11	Canadian Aboriginal	74
Arrows	38 98 99 110	Carian	1025
Avestan	1030	Cham	143
Balinese	129	Cherokee	75
Bamum	145	CJK	49 54 55 56 60 62 66 78 81 124 2001 2002
Bengali	17		
Bidirectional	202 203	Combining	7 35 65 117 270 271
Block elements	45	Compatibility	53 56 62 66
BMP	300 301 302 (299)	Control pictures	41
Box drawing	44	Coptic	9 119

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~~point

.

Block name	from to		
BASIC LATIN	0020-007E	LATIN EXTENDED-B	0180-024F
LATIN-1 SUPPLEMENT	00A0-00FF	IPA EXTENSIONS	0250-02AF
LATIN EXTENDED-A	0100-017F	SPACING MODIFIER LETTERS	02B0-02FF
		COMBINING DIACRITICAL MARKS	0300-036F

GREEK AND COPTIC	0370-03FF
CYRILLIC	0400-04FF
CYRILLIC SUPPLEMENT	0500-052F
ARMENIAN	0530-058F
HEBREW	0590-05FF
ARABIC	0600-06FF
SYRIAC	0700-074F
ARABIC SUPPLEMENT	0750-077F
THAANA	0780-07BF
NKO	07C0-07FF
DEVANAGARI	0900-097F
BENGALI	0980-09FF
GURMUKHI	0A00-0A7F
GUJARATI	0A80-0AFF
ORIYA	0B00-0B7F
TAMIL	0B80-0BFF
TELUGU	0C00-0C7F
KANNADA	0C80-0CFF
MALAYALAM	0D00-0D7F
SINHALA	0D80-0DFF
THAI	0E00-0E7F
LAO	0E80-0EFF
TIBETAN	0F00-0FFF
MYANMAR	1000-109F
GEORGIAN	10A0-10FF
HANGUL JAMO	1100-11FF
ETHIOPIC	1200-137F
ETHIOPIC SUPPLEMENT	1380-139F
CHEROKEE	13A0-13FF
UNIFIED CANADIAN ABORIGINAL SYLLABICS	1400-167F
OGHAM	1680-169F
RUNIC	16A0-16FF
TAGALOG	1700-171F
HANUNOO	1720-173F
BUHID	1740-175F
TAGBANWA	1760-177F
KHMER	1780-17FF
MONGOLIAN	1800-18AF
LIMBU	1900-194F
TAI LE	1950-197F
NEW TAI LUE (Xishuang Banna Dai)	1980-19DF
KHMER SYMBOLS	19E0-19FF
BUGINESE	1A00-1A1F
LANNA (Old Tai Lue)	1A20-1AAF
BALINESE	1B00-1B7F
SUNDANESE	1B80-1BBF
LEPCHA	1C00-1C4F
OL CHIKI	1C50-1C7F
MEITEI MAYEK	1C80-1CCF
PHONETIC EXTENSIONS	1D00-1D7F
PHONETIC EXTENSIONS SUPPLEMENT	1D80-1DBF
COMBINING DIACRITICAL MARKS SUPPLEMENT	1DC0-1DFF
LATIN EXTENDED ADDITIONAL	1E00-1EFF
GREEK EXTENDED	1F00-1FFF
GENERAL PUNCTUATION	2000-206F
SUPERSCRIPTS AND SUBSCRIPTS	2070-209F
CURRENCY SYMBOLS	20A0-20CF
COMBINING DIACRITICAL MARKS FOR SYMBOLS	20D0-20FF
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 ALPHANUMERICS	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
MISCELLANEOUS MATHEMATICAL SYMBOLS-B	2980-29FF
SUPPLEMENTAL MATHEMATICAL OPERATORS	2A00-2AFF
MISCELLANEOUS SYMBOLS AND ARROWS	2B00-2BFF
GLAGOLITIC	2C00-2C5F
LATIN EXTENDED-C	2C60-2C7F
COPTIC	2C80-2CFF
GEORGIAN SUPPLEMENT	2D00-2D2F
TIFINAGH	2D30-2D7F
ETHIOPIC EXTENDED	2D80-2DDF
CYRILLIC EXTENDED-A	2DE0-2DFF
SUPPLEMENTAL PUNCTUATION	2E00-2E7F
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
CJK STROKES	31C0-31EF
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
VAI	A500-A63F
CYRILLIC EXTENDED-B	A640-A69F
BAMUM	A6A0-A6FF
MODIFIER TONE LETTERS	A700-A71F
LATIN EXTENDED-D	A720-A7FF
SYLOTI NAGRI	A800-A82F
PHAGS-PA	A840-A87F
SAURASHTRA	A880-A8DF
KAYAH LI	A900-A92F
REJANG	A930-A95F
HANGUL JAMO EXTENDED-A	A960-A97F
CHAM	AA00-AA5F
TAI VIET	AA80-AADF
HANGUL SYLLABLES	AC00-D7A3
HANGUL JAMO EXTENDED-B	D7B0-D7FF
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	SMALL FORM VARIANTS	FE50-FE6F
VERTICAL FORMS	FE10-FE1F	ARABIC PRESENTATION FORMS-B	FE70-FEFE
COMBINING HALF MARKS	FE20-FE2F	HALFWIDTH AND FULLWIDTH FORMS	FF00-FFEF
CJK COMPATIBILITY FORMS	FE30-FE4F	SPECIALS	FFF0-FFFD

NOTE – The parenthetical annotation located in some block names is not part of these names.

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~~point.

Block name	from to		
LINEAR B SYLLABARY	10000-1007F	PHOENICIAN	10900-1091F
LINEAR B IDEOGRAMS	10080-100FF	LYDIAN	10920-1093F
AEGEAN NUMBERS	10100-1013F	KHAROSHTHI	10A00-10A5F
ANCIENT GREEK NUMBERS	10140-1018F	AVESTAN	10B00-10B3F
ANCIENT SYMBOLS	10190-101CF	CUNEIFORM	12000-123FF
PHAISTOS DISC	101D0-101FF	CUNEIFORM NUMBERS AND PUNCTUATION	12400-1247F
LYCIAN	10280-1029F	EGYPTIAN HIEROGLYPHS	13000-1342F
CARIAN	102A0-102DF	BYZANTINE MUSICAL SYMBOLS	1D000-1D0FF
OLD ITALIC	10300-1032F	MUSICAL SYMBOLS	1D100-1D1FF
GOTHIC	10330-1034F	ANCIENT GREEK MUSICAL NOTATION	1D200-1D24F
UGARITIC	10380-1039F	TAI XUAN JING SYMBOLS	1D300-1D35F
OLD PERSIAN	103A0-103DF	COUNTING ROD NUMERALS	1D360-1D37F
DESERET	10400-1044F	MATHEMATICAL ALPHANUMERIC SYMBOLS	1D400-1D7FF
SHAVIAN	10450-1047F	MAHJONG TILES	1F000-1F02F
OSMANYA	10480-104AF	DOMINO TILES	1F030-1F09F
CYPRIOT SYLLABARY	10800-1083F		

A.2.3 Blocks in the SIP

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

Block name	from to
CJK UNIFIED IDEOGRAPHS EXTENSION B	20000-2A6DF
CJK UNIFIED IDEOGRAPHS EXTENSION C	2A700-2B77F
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~~point.

Block name	from to
TAGS	E0000-E007F
VARIATION SELECTORS SUPPLEMENT	E0100-E01EF

A.3 Fixed collections of the whole UCS (except Unicode collections)

The following fixed collections (see 4.24~~4.22~~) contain the whole UCS assigned character content as it was when they were created. The Unicode collections are described in A.1~~A.6~~.

A.3.1 301 BMP-AMD.7

The fixed collection 301 BMP-AMD.7 is specified below. 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.

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

Plane 00

Row~~s~~	~~Positions (cells)~~Values within row
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-F8FF
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 fixed collection 299 BMP FIRST EDITION has been reserved to identify 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:

~~r~~Row~~s~~	~~p~~Values within row~~ositions~~
05	31-56 59-5F 61-87 89 B0-B9 BB-C3 D0-EA F0-F4
0F	[no ~~positions~~values]
1E	00-9A A0-F9
20	00-2E 30-46 6A-70 74-8E A0-AA D0-E1
AC-D7	[no ~~position~~value~~ss~~]

and by including an additional entry:

~~r~~Row~~s~~	~~positions~~ Values within row
34-4D	3400-4DFF

for the code ~~position ranges~~point values 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~~ points as indicated for each row or contiguous series of rows.

Plane 00

Row ~~Rows~~	Values within row ~~Positions (cells)~~
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.3.4 340 COMBINED FIRST EDITION

The fixed collection 340 COMBINED FIRST EDITION is specified below. It comprises only those coded characters that were in the First Edition of 10646:2003 and consists of collections from A.1~~A.1~~ and A.3~~A.3~~ and several ranges of code ~~positions~~points. The collection list is arranged by planes as follows.

Plane 00

Collection number and name	
302	BMP SECOND EDITION
98	SUPPLEMENTAL ARROWS-A
99	SUPPLEMENTAL ARROWS-B
100	MISCELLANEOUS MATHEMATICAL SYMBOLS-B
101	SUPPLEMENTAL MATHEMATICAL OPERATORS
102	KATAKANA PHONETIC EXTENSIONS
103	VARIATION SELECTORS
108	KHMER SYMBOLS
111	YIJING HEXAGRAM SYMBOLS

Row ~~Row~~	Values within row ~~Positions (cells)~~
02	20-21 34-36 AE-AF EF-FF
03	4F-57 5D-5F 63-6F D8-D9 F4-FB
04	8A-8B C5-C6 C9-CA CD-CE
05	00-0F
06	00-03 0D-15 56-58 6E-6F EE-EF FF
07	2D-2F 4D-4F B1
09	04 BD
0A	01 03 8C E1-E3 F1
0B	35 71 F3-FA
0C	BC-BD
10	F7-F8

17	00-0C 0E-14 20-36 40-53 60-6C 6E-70 72-73 DD F0-F9
19	00-1C 20-2B 30-3B 40 44-4F 50-6D 70-74
1D	00-6B
20	47 4E-54 57 5F-63 71 B0-B1 E4-EA
21	3B 3D-4B F4-FF
22	F2-FF
23	7C 9B-D0
24	EB-FF
25	96-9F F8-FF
26	14-17 72-7D 80-91 A0-A1
27	68-75 D0-EB
2B	00-0D
30	3B-3D 95-96 9F-A0 FF
32	1D-1E 50-5F 7C-7D B1-BF CC-CF
33	77-7A DE-DF FF
A4	A2-A3 B4 C1 C5
FA	30-6A
FD	FC-FD
FE	45-48 73
FF	5F-60

Plane 01

Collection number and name

1003	DESERET
1011	SHAVIAN

Row	Values within row
~~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	80-9D A0-A9
08	00-05 08 0A-35 37-38 3C 3F
D0	00-F5
D1	00-26 2A-DD
D3	00-56
D4	00-54 56-9C 9E-9F A2 A5-A6 A9-AC AE-B9 BB BD-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 02

Row	Values within row
~~Row~~	~~Positions (cells)~~
00-A6	0000-A6D6
F8-FA	F800-FA1D

Plane 0E

Collection number and name

3003	VARIATION SELECTORS SUPPLEMENT

Row	Values within row
~~Row~~	~~Positions (cells)~~
00	01 20-7F

Plane 0F

Row	Values within row
~~Row~~	~~Positions (cells)~~
00-FF	0000-FFFD

Plane 10

Row	Values within row
~~Row~~	~~Positions (cells)~~
00-FF	0000-FFFD

A.4 CJK collections

A.4.1 370 IICORE

The fixed collection 370 IICORE is the International Core subset of the CJK UNIFIED IDEOGRAPHS-2001 collection.

NOTE 1 – Given its large size (9810 characters) and the large number of sparse ranges, the collection is not speci-fied by ~~Rows/Positions~~code point ranges in this document but instead by a linked content.

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 IICORE characters; each containing the following information in fixed length field.

- 1st field: BMP or SIP code ~~position~~ point (`0hhhh`), (`2hhhh`), normative.
- 2nd field: Hanzi G usage identifier (`G0a`), (`G1a`), (`G3a`), (`G5a`), (`G7a`), (`G8a`), (`G9a`), or (`GEa`), informa-tive.
- 3rd field: Hanzi T usage identifier (`T1a`), (`T2a`), (`T3a`), (`T4a`), (`T5a`) or (`TFa`), informative.
- 4th field: Kanji J usage identifier (`J1A`), in-formative.
- 5th field: Hanzi H usage identifier (`H1a`), in-formative.
- 6th field: Hanja K usage identifier (`K0a`), (`K1a`), (`K2a`) or (`K3a`), informative.
- 7th field: Hanzi M (for Macao SAR) usage identifier (`M1a`), informative.
- 8th field: Hanja KP usage identifier (`P0a`), informative.
- 9th field: General category, informative (`A`, `B` or `C` in decreasing order of priority).

The format definition uses '`h`' as a hexadecimal unit and '`a`' as an enumerated unit for letters from 'A' to 'G'. Uppercase characters and digits between parentheses appear as shown.

NOTE 2 – The usage information provided in this subclause describes the usage and priority level of individual IICORE char-acters in the context of each source (G, T, J, H, K, M, and KP). This should not be confused with the source references for CJK Ideographs in 23~~27~~ which establish the identity of all CJK Ideographs.

Click on this highlighted text to access the reference file.

NOTE 3 – The content is also available as a separate viewable file in the same file directory as this document. The file is named: "IICORE.txt".

A.4.2 371 JIS2004 IDEOGRAPHICS EXTENSION

The fixed collection 371 JIS2004 IDEOGRAPHICS EXTENSION consists of all level 3 and level 4 CJK characters defined in JIS X 0213:2004.

NOTE 1 – Given its large size (3695 characters) and the large number of sparse ranges, the collection is not specified by ~~Rows/Positions~~code point ranges in this document but instead by a linked content.

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 3-lines header, as many lines as characters in the collection; each containing the following information in fixed length field:

- BMP or SIP code ~~position~~ point (`0hhhh`), (`2hhhh`), normative.

The format definition uses '`h`' as a hexadecimal unit. Digits between parentheses 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: "JIEx.txt".

A.4.3 372 JAPANESE IDEOGRAPHICS SUPPLEMENT

The fixed collection 372 JAPANESE IDEOGRAPHICS SUPPLEMENT consists of all CJK characters de-fined in JIS X 0212:1990. It contains 5801 characters.

NOTE – 2742 characters are common between the collections 371 and 372.

The code ~~positions~~ points of this collection are identified by the J1 Kanji J sources in the Source Refer-ence file for CJK Unified Ideographs (CJKU_SR.txt). See 23.1~~27.1~~ for further details.

A.5 Other collections

The collections specified within this clause address the referencing need of users community. Characters may be from different writing systems and may be coded in different planes. It includes collection for users community from Lithuania, Japan and Europe as a whole.

NOTE – The acronym MES used in collection names below indicates Multilingual European Subset.

A.5.1 281 MES-1

The fixed collection 281 MES-1 is specified by the following ranges of code ~~positions~~ points as indicated for each row.

Plane 00

Row ~~Rows~~	Values within row ~~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.5.2 282 MES-2

The fixed collection 282 MES-2 is specified by the following ranges of code ~~positions~~ points as indicated for each row.

Plane 00

Row ~~Rows~~	Values within row ~~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.5.3 283 MODERN EUROPEAN SCRIPTS

The collection 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.5.4 284 CONTEMPORARY LITHUANIAN LETTERS

The fixed extended collection 284 CONTEMPORARY LITHUANIAN LETTERS is defined as follows.

Plane 00

Row ~~Row~~	Values within row ~~Positions (cells)~~
00	41-50 52-56 59-5A 61-70 72-76 79-7A C0-C1 C3 C8-C9 CC-CD D1-D3 D5 D9-DA DD E0-E1 E3 E8-E9 F1-F3 F5 F9-FA FD
01	04-05 0C-0D 16-19 28 2E-2F 60-61 68-6B 72-73 7D-7E
1E	BC-BD F8-F9

UCS Sequence Indentifiers

<0104, 0301> <0105, 0301> <0104, 0303> <0105, 0303> <0118, 0301> <0119, 0301> <0118, 0303> <0119, 0303> <0116, 0301> <0117, 0301> <0116, 0303> <0117, 0303> <0069, 0307, 0300> <0069, 0307, 0301> <0069, 0307, 0303> <012E, 0301> <012F, 0307, 0301> <012E, 0303> <012F, 0307, 0303> <004A, 0303> <006A, 0307, 0303> <004C, 0303> <006C, 0303> <004D, 0303> <006D, 0303> <0052, 0303> <0072, 0303> <0172, 0301> <0173, 0301> <0172, 0303> <0173, 0303> <016A, 0301> <016B, 0301> <016A, 0303> <016B, 0303>

A.5.5 285 BASIC JAPANESE

The fixed collection 285 BASIC JAPANESE is a core Japanese subset. Its 6884 characters are identified by:

- All J0 Kanji J sources in the Source Reference file for CJK Unified Ideographs (CJKU_SR.txt). See 23.1~~27.1~~ for further details.
- Ranges of code ~~positions~~ points arranged by planes:

Plane 00

Row ~~Row~~	Values within row ~~Positions (cells)~~
00	20-7E A2 A3 A5 A7-A8 AC B0-B1 B4 B6 D7 F7
03	91-A1 A3-A9 B1-C1 C3-C9
04	01 10-4F 51
20	10 14 16 18-19 1C-1D 20-21 25-26 30 32-33 3B 3E
21	03 2B 90-93 D2 D4
22	00 02-03 07-08 0B 12 1A 1D-1E 20 27-2C 34-35 3D 52 60-61 66-67 6A-6B 82-83 86-87 A5
23	12
25	00-03 0C 0F-10 13-14 17-18 1B-1D 20 23-25 28 2B-2C 2F-30 33-34 37-38 3B-3C 3F 42 4B A0-A1 B2-B3 BC-BD C6-C7 CB CE-CF EF
26	05-06 40 42 6A 6D 6F
30	00-03 05-15 1C 41-93 9B-9E A1-F6 FB-FE

A.5.6 286 JAPANESE NON IDEOGRAPHICS EXTENSION

The fixed collection 286 JAPANESE NON IDEOGRAPHICS EXTENSION is a Japanese subset which completes JIS X 0213 non-ideographic repertoire in combination with either 285 BASIC JAPANESE or 287 COMMON JAPANESE. Its 631 characters are identified by the following ranges of code ~~positions~~ points arranged by planes:

Plane 00

Row ~~Row~~	Values within row ~~Positions (cells)~~
00	A0-A1 A4 A6 A9-AB AD-AF B2-B3 B7-D6 D8-F6 F8-FF
01	00-09 0C-0F 11-13 18-1D 24-25 27 2A-2B 34-35 39-3A 3D-3E 41-44 47-48 4B-4D 50-55 58-65 6A-71 79-7E 93 C2 CD-CE D0-D2 D4 D6 D8 DA DC F8-F9 FD
02	50-5A 5C 5E-61 64-68 6C-73 75 79-7B 7D-7E 81-84 88-8E 90-92 94-95 98 9D A1-A2 C7-C8 CC D0-D1 D8-D9 DB DD-DE E5-E9
03	00-04 06 08 0B-0C 0F 18-1A 1C-20 24-25 29-2A 2C 2F-30 34 39-3D 61 C2
1E	3E-3F
1F	70-73
20	13 22 3C 3F 42 47-49 51 AC
21	0F 13 16 21 27 35 53-55 60-6B 70-7B 94 96-99 C4 E6-E9

22	05 09 13 1F 25-26 2E 43 45 48 62 76-77 84-85 8A-8B 95-97 BF DA-DB
23	05-06 18 BE-CC CE
24	23 60-73 D0-E9 EB-FE
25	B1 B6-B7 C0-C1 C9 D0-D3 E6
26	00-03 0E 16-17 1E 60-69 6B-6C 6E
27	13 56 76-7F
29	34-35 BF FA-FB
30	16-19 1D 1F-20 33-35 3B-3D 94-96 9A 9F-A0 F7-FA FF
31	F0-FF
32	31-32 39 51-5F A4-A8 B1-BF D0-E3 E5 E9 EC-ED FA
33	03 0D 14 18 22-23 26-27 2B 36 3B 49-4A 4D 51 57 7B-7E 8E-8F 9C-9E A1 C4 CB CD
FE	45-46
FF	5F-60

A.5.7 287 COMMON JAPANESE

The fixed collection 287 COMMON JAPANESE is a core Japanese subset containing 7493 characters. It includes a fixed collection from A.5 and several ranges of code ~~positions~~points.

Planes 00-10

Collection number and name

285 BASIC JAPANESE

Plane 00

Row	Values within row
~~Row~~	~~Positions (cells)~~
20	15
21	16 21 60-69 70-79
22	11 1F 25 2E BF
24	60-73
30	1D 1F
32	31-32 39 A4-A8
33	03 0D 14 18 22-23 26-27 2B 36 3B 49-4A 4D 51 57 7B-7E 8E-8F 9C-9E A1 C4 CD
4E	28 E1 FC
4F	00 03 39 56 8A 92 94 9A C9 CD FF
50	1E 22 40 42 46 70 94 D8 F4
51	4A 64 9D BE EC
52	15 9C A6 AF C0 DB
53	00 07 24 72 93 B2 DD
54	8A 9C A9 FF
55	86
57	59 65 AC C7-C8
58	9E B2
59	0B 53 5B 5D 63 A4 BA
5B	56 C0 D8 EC
5C	1E A6 BA F5
5D	27 42 53 6D B8-B9 D0
5F	21 34 45 67 B7 DE
60	5D 85 8A D5 DE F2
61	11 20 30 37 98
62	13 A6
63	F5
64	60 9D CE
65	4E
66	00 09 15 1E 24 2E 31 3B 57 59 65 73 99 A0 B2 BF FA-FB
67	0E 66 BB C0
68	01 44 52 C8 CF
69	68 98 E2
6A	30 46 6B 73 7E E2 E4
6B	D6
6C	3F 5C 6F 86 DA
6D	04 6F 87 96 AC CF F2 F8 FC
6E	27 39 3C 5C BF
6F	88 B5 F5
70	05 07 28 85 AB BB
71	04 0F 46-47 5C C1 FE
72	B1 BE
73	24 77 BD C9 D2 D6 E3 F5
74	07 26 29-2A 2E 62 89 9F
75	01 2F 6F
76	82 9B-9C 9E A6
77	46
78	21 4E 64 7A
79	30 94 9B
7A	D1 E7 EB
7B	9E
7D	48 5C A0 B7 D6
7E	52 8A
7F	47 A1
83	01 62 7F C7 F6
84	48 B4 DC
85	53 59 6B B0
88	07 F5
89	1C
8A	12 37 79 A7 BE DF F6
8B	53 7F
8C	F0 F4
8D	12 76
8E	CF
90	67 DE
91	15 27 D7 DA DE E4-E5 ED-EE
92	06 0A 10 39-3A 3C 40 4E 51 59 67 77-78 88 A7 D0 D3 D5 D7 D9 E0 E7 F9 FB FF
93	02 1D-1E 21 25 48 57 70 A4 C6 DE F8
94	31 45 48
95	92
96	9D AF
97	33 3B 43 4D 4F 51 55
98	57 65
99	27 9E
9A	4E D9 DC
9B	72 75 8F B1 BB
9C	00
9D	6B 70
9E	19 D1
F9	29 DC
FA	0E-2D
FF	01-5E 61-9F E0-E5

A.6 Unicode collections

These collections correspond to various versions of the Unicode Standard. They include characters from the BMP as well as Supplementary planes.

NOTE – Unicode 2.0 corresponds to collection 301. Unicode 2.1 adds the code ~~positions~~ points 20AC EURO SIGN and FFFC OBJECT REPLACEMENT CHARACTER to the collection 301. Unicode 3.0 corresponds to collection 302.

A.6.1 303 UNICODE 3.1

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

Plane 00
Collection number and name
302 BMP SECOND EDITION

Row Values within row
~~Row Positions (cells)~~
03 F4-F5

Plane 01
Row Values within row
~~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 02
Row Values within row
~~Row Positions (cells)~~
00-A6 0000-A6D6
F8-FA F800-FA1D

Plane 0E
Row Values within row
~~Row Positions (cells)~~
00 01 20-7F

Plane 0F
Row Values within row
~~Row Positions (cells)~~
00-FF 0000-FFFD

Plane 10
Row Values within row
~~Row Positions (cells)~~
00-FF 0000-FFFD

A.6.2 304 UNICODE 3.2

The fixed collection 304 UNICODE 3.2 consists of fixed collections from A.1~~A.1~~ and A.1~~A.6~~ and several ranges of code ~~positions~~ points arranged by planes as follows.

Planes 00-10
Collection number and name
303 UNICODE 3.1

Plane 00
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

Row	Values within row
~~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.6.3 305 UNICODE 4.0

The fixed collection 305 UNICODE 4.0 is identical to the fixed collection 340 COMBINED FIRST EDITION.

A.6.4 306 UNICODE 4.1

The fixed collection 306 UNICODE 4.1 consists of a fixed collection from A.1~~A.6~~ and several ranges of code ~~positions~~points. The collection list is arranged by planes as follows.

Plane 00-10
Collection number and name
305 UNICODE 4.0

Plane 00

Row	Values within row
~~Row~~	~~Positions (cells)~~
02	37-41
03	58-5C FC-FF
04	F6-F7
05	A2 C5-C7
06	0B 1E 59-5E
07	50-6D
09	7D CE
0B	B6 E6
0F	D0-D1
10	F9-FA FC
12	07 47 87 AF CF EF
13	0F 1F 47 5F-60 80-99
19	80-A9 B0-C9 D0-D9 DE-DF
1A	00-1B 1E-1F
1D	6C-C3
20	55-56 58-5E 90-94 B2-B5 EB
21	3C 4C
23	D1-DB
26	18 7E-7F 92-9C A2-B1
27	C0-C6
2B	0E-13
2C	00-2E 30-5E 80-EA F9-FF
2D	00-25 30-65 6F 80-96 A0-A6 A8-AE B0-B6 B8-BE C0-C6 C8-CE D0-D6 D8-DE
2E	00-17 1C-1D
31	C0-CF
32	7E
9F	A6-BB
A7	00-16
A8	00-2B
FA	70-D9
FE	10-19

Plane 01

Row	Values within row
~~Row~~	~~Positions (cells)~~
01	40-8A
03	A0-C3 C8-D5
0A	00-03 05-06 0C-13 15-17 19-33 38-3A 3F-47 50-58
D2	00-45
D6	A4-A5

A.6.5 307 UNICODE 5.0

The fixed collection 307 UNICODE 5.0 consists of a fixed collection from A.1~~A.6~~ and several ranges of code ~~positions~~points. The collection list is arranged by planes as follows.

Plane 00-10
Collection number and name
306 UNICODE 4.1

Plane 00

Row	Values within row	~~Row~~	~~Positions (cells)~~

02	42-4F
03	7B-7D
04	CF FA-FF
05	10-13 BA
07	C0-FA
09	7B-7C 7E-7F
0C	E2-E3 F1-F2
1B	00-4B 50-7C
1D	C4-CA FE-FF
20	EC-EF
21	4D-4E 84
23	DC-E7
26	B2
27	C7-CA
2B	14-1A 20-23
2C	60-6C 74-77
A7	17-1A 20-21
A8	40-77

Plane 01

Row	Values within row
~~Row~~	~~Positions (cells)~~
09	00-19 1F
20-22	2000-22FF
23	00-6E
24	00-62 70-73
D3	60-71
D7	CA-CB

A.6.6 308 UNICODE 5.1

The fixed collection UNICODE 5.1 is arranged by planes as follows.

Plane 00

Row	Values within row
~~Row~~	~~Positions (cells)~~
00	20-7E A0-FF
01-02	0100-02FF
03	00-77 7A-7E 84-8A 8C 8E-A1 A3-FF
04	00-FF
05	00-23 31-56 59-5F 61-87 89-8A 91-C7 D0-EA F0-F4
06	00-03 06-1B 1E-5E 60-FF
07	00-0D 0F-4A 4D-B1 C0-FA
09	01-39 3C-4D 50-54 58-72 7B-7F 81-83 85-8C 8F-90 93-A8 AA-B0 B2 B6-B9 BC-C4 C7-C8 CB-CE D7 DC-DD DF-E3 E6-FA
0A	01-03 05-0A 0F-10 13-28 2A-30 32-33 35-36 38-39 3C 3E-42 47-48 4B-4D 51 59-5C 5E 66-75 81-83 85-8D 8F-91 93-A8 AA-B0 B2-B3 B5-B9 BC-C5 C7-C9 CB-CD D0 E0-E3 E6-EF F1
0B	01-03 05-0C 0F-10 13-28 2A-30 32-33 35-39 3C-44 47-48 4B-4D 56-57 5C-5D 5F-63 66-71 82-83 85-8A 8E-90 92-95 99-9A 9C 9E-9F A3-A4 A8-AA AE-B9 BE-C2 C6-C8 CA-CD D0 D7 E6-FA
0C	01-03 05-0C 0E-10 12-28 2A-33 35-39 3D-44 46-48 4A-4D 55-56 58-59 60-63 66-6F 78-7F 82-83 85-8C 8E-90 92-A8 AA-B3 B5-B9 BC-C4 C6-C8 CA-CD D5-D6 DE E0-E3 E6-EF F1-F2
0D	02-03 05-0C 0E-10 12-28 2A-39 3D-44 46-48 4A-4D 57 60-63 66-75 79-7F 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-A8 AD-B9 BB-BD C0-C4 C6 C8-CD D0-D9 DC-DD
0F	00-47 49-6C 71-8B 90-97 99-BC BE-CC CE-D4
10	00-8A A0-C5 D0-FC
11	00-59 5F-A2 A8-F9
12	00-48 4A-4D 50-56 58 5A-5D 60-88 8A-8D 90-B0 B2-B5 B8-BE C0 C2-C5 C8-D6 D8-FF
13	00-10 12-15 18-5A 5F-7C 80-99 A0-F4
14-15	1401-15FF
16	00-76 80-9C A0-F0
17	00-0C 0E-14 20-36 40-53 60-6C 6E-70 72-73 80-DD E0-E9 F0-F9
18	00-0E 10-19 20-77 80-AA
19	00-1C 20-2B 30-3B 40 44-6D 70-74 80-A9 B0-C9 D0-D9 DE-FF
1A	00-1B 1E-7B 7F-89 90-99 A0-AD
1B	00-4B 50-7C 80-AA AE-B9
1C	00-37 3B-49 4D-7F
1D	00-E6 FE-FF
1E	00-FF
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	00-64 6A-71 74-8E 90-94 A0-B5 D0-F0
21	00-4F 53-88 90-FF
22	00-FF
23	00-E7
24	00-26 40-4A 60-FF
25	00-FF
26	00-9D A0-BC C0-C3
27	01-04 06-09 0C-27 29-4B 4D 4F-52 56 58-5E 61-94 98-AF B1-BE C0-CA CC D0-FF
28-2A	2800-2AFF
2B	00-4C 50-54
2C	00-2E 30-5E 60-6F 71-7D 80-EA F9-FF
2D	00-25 30-65 6F 80-96 A0-A6 A8-AE B0-B6 B8-BE C0-C6 C8-CE D0-D6 D8-DE E0-FF
2E	00-1F 2A 2C 2E-2F 34 38 3B 40-49 80-99 9B-F3
2F	00-D5 F0-FB
30	00-3F 41-96 99-FF
31	05-2D 31-8E 90-B7 C0-E3 F0-FF
32	00-1E 20-43 50-FE
33	00-FF
34-4C	3400-4CFF
4D	00-B5 C0-FF
4E-9F	4E00-9FC3
A0-A3	A000-A3FF
A4	00-8C 90-C6
A5	00-FF
A6	00-2B 40-5F 62-73 7C-97
A7	00-8C FB-FF
A8	00-2B 40-77 80-C4 CE-D9
A9	00-53 5F
AA	00-36 40-4D 50-59 5C-5F
AC-D7	AC00-D7A3
E0-F8	E000-F8FF
F9	00-FF
FA	00-2D 30-6A 70-D9
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-FD
FE	00-19 20-26 30-52 54-66 68-6B 70-74 76-FC FF
FF	01-BE C2-C7 CA-CF D2-D7 DA-DC E0-E6 E8-EE F9-FD

Plane 01

Row ~~Row~~	Values within row ~~Positions (cells)~~
00	00-0B 0D-26 28-3A 3C-3D 3F-4D 50-5D 80-FA
01	00-02 07-33 37-8A 90-9B D0-FD
02	80-9C A0-D0
03	00-1E 20-23 30-4A 80-9D 9F-C3 C8-D5
04	00-9D A0-A9
08	00-05 08 0A-35 37-38 3C 3F
09	00-19 1F-39 3F
0A	00-03 05-06 0C-13 15-17 19-33 38-3A 3F-47 50-58
20-22	2000-22FF
23	00-6E
24	00-62 70-73
D0	00-F5
D1	00-26 29-DD
D2	00-45
D3	00-56 60-71
D4	00-54 56-9C 9E-9F A2 A5-A6 A9-AC AE-B9 BB BD-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-A5 A8-FF
D7	00-CB CE-FF
F0	00-2B 30-93

Plane 02

Row ~~Row~~	Values within row ~~Positions (cells)~~
00-A6	0000-A6D6
F8-FA	F800-FA1D

Plane 0E

Row ~~Row~~	Values within row ~~Positions (cells)~~
00	01 20-7F
01	00-EF

Plane 0F

Row ~~Row~~	Values within row ~~Positions (cells)~~
00-FF	0000-FFFD

Plane 10

Row ~~Row~~	Values within row ~~Positions (cells)~~
00-FF	0000-FFFD

NOTE – The collection 309 UNICODE 5.1 can also be determined by using another fixed collection from A.1~~A.6~~ and several ranges of code ~~positions~~points.

Plane 00-10

Collection number and name	
308	UNICODE 5.0

Plane 00

Row	~~Positions (cells~~Values within row~~)~~
03	70-73 76-77 CF
04	87
05	14-23
06	06-0A 16-1A 3B-3F
07	6E-7F
09	71-72
0A	51 75
0B	44 62-63 D0
0C	3D 58-59 62-63 78-7F
0D	3D 44 62-63 70-75 79-7F
0F	6B-6C CE D2-D4
10	22 28 2B 33-35 3A-3F 5A-8A
18	AA
1A	20-7b 7F-89 90-99 A0-AD
1B	80-AA AE-B9
1C	00-37 3B-49 4D-7F
1D	CB-E6
1E	9C-9F FA-FF
20	64 F0
21	4F 85-88
26	9D B3-BC C0-C3
27	CC EC-EF
2B	1B-1F 24-4C 50-54
2C	6D-6F 71-73 78-7D
2D	E0-FF
2E	18-1B 1E-1F 2A 2C 2E-2F 34 38 3B 40-49
31	2D D0-E3

9F	BC-C3
A5	00-FF
A6	00-2B 40-5F 62-73 7C-97
A7	1B-1F 22-8C FB-FF
A8	80-C4 CE-D9
A9	00-53 5F
AA	00-36 40-4D 50-59 5C-5F
FE	24-26

Plane 01

Row	~~Positions (cells)~~Values within row
01	90-9B D0-FD
02	80-9C A0-D0
09	20-39 3F
D1	29
F0	00-2B 30-93

~~Annex B~~
~~(normative)~~
~~List of combining characters~~

~~The characters in the collections:~~

- ~~COMBINING DIACRITICAL MARKS (0300-036F),~~
- ~~COMBINING DIACRITICAL MARKS SUPPLEMENT (1DC0-1DFF),~~
- ~~COMBINING DIACRITICAL MARKS FOR SYMBOLS (20D0-20FF),~~
- ~~CYRILLIC EXTENDED-A (2DE0-2DFF),~~
- ~~VARIATION SELECTORS (FE00-FE0F),~~
- ~~COMBINING HALF MARKS (FE20-FE2F), and~~
- ~~VARIATION SELECTORS SUPPLEMENT (E0100-E01EF)~~

~~are combining characters. In addition, the following characters are combining characters.~~

~~0483 COMBINING CYRILLIC TITLO~~
~~0484 COMBINING CYRILLIC PALATALIZATION~~
~~0485 COMBINING CYRILLIC DASIA PNEUMATA~~
~~0486 COMBINING CYRILLIC PSILI PNEUMATA~~
~~0487 COMBINING CYRILLIC POKRYTIE~~
~~0488 COMBINING CYRILLIC HUNDRED THOUSANDS SIGN~~
~~0489 COMBINING CYRILLIC MILLIONS SIGN~~
~~0591 HEBREW ACCENT ETNAHTA~~
~~0592 HEBREW ACCENT SEGOL~~
~~0593 HEBREW ACCENT SHALSHELET~~
~~0594 HEBREW ACCENT ZAQEF QATAN~~
~~0595 HEBREW ACCENT ZAQEF GADOL~~
~~0596 HEBREW ACCENT TIPEHA~~
~~0597 HEBREW ACCENT REVIA~~
~~0598 HEBREW ACCENT ZARQA~~
~~0599 HEBREW ACCENT PASHTA~~
~~059A HEBREW ACCENT YETIV~~
~~059B HEBREW ACCENT TEVIR~~
~~059C HEBREW ACCENT GERESH~~
~~059D HEBREW ACCENT GERESH MUQDAM~~
~~059E HEBREW ACCENT GERSHAYIM~~
~~059F HEBREW ACCENT QARNEY PARA~~
~~05A0 HEBREW ACCENT TELISHA GEDOLA~~
~~05A1 HEBREW ACCENT PAZER~~
~~05A2 HEBREW ACCENT ATNAH HAFUKH~~
~~05A3 HEBREW ACCENT MUNAH~~
~~05A4 HEBREW ACCENT MAHAPAKH~~
~~05A5 HEBREW ACCENT MERKHA~~
~~05A6 HEBREW ACCENT MERKHA KEFULA~~
~~05A7 HEBREW ACCENT DARGA~~
~~05A8 HEBREW ACCENT QADMA~~
~~05A9 HEBREW ACCENT TELISHA QETANA~~
~~05AA HEBREW ACCENT YERAH BEN YOMO~~
~~05AB HEBREW ACCENT OLE~~
~~05AC HEBREW ACCENT ILUY~~
~~05AD HEBREW ACCENT DEHI~~
~~05AE HEBREW ACCENT ZINOR~~
~~05AF HEBREW MARK MASORA CIRCLE~~
~~05B0 HEBREW POINT SHEVA~~
~~05B1 HEBREW POINT HATAF SEGOL~~
~~05B2 HEBREW POINT HATAF PATAH~~
~~05B3 HEBREW POINT HATAF QAMATS~~
~~05B4 HEBREW POINT HIRIQ~~
~~05B5 HEBREW POINT TSERE~~
~~05B6 HEBREW POINT SEGOL~~

~~05B7 HEBREW POINT PATAH~~
~~05B8 HEBREW POINT QAMATS~~
~~05B9 HEBREW POINT HOLAM~~
~~05BA HEBREW POINT HOLAM HASER FOR VAV~~
~~05BB HEBREW POINT QUBUTS~~
~~05BC HEBREW POINT DAGESH OR MAPIQ~~
~~05BD HEBREW POINT METEG~~
~~05BF HEBREW POINT RAFE~~
~~05C1 HEBREW POINT SHIN DOT~~
~~05C2 HEBREW POINT SIN DOT~~
~~05C4 HEBREW MARK UPPER DOT~~
~~05C5 HEBREW MARK LOWER DOT~~
~~05C7 HEBREW POINT QAMATS QATAN~~
~~0610 ARABIC SIGN SALLALLAHOU ALAYHE WASALLAM~~
~~0611 ARABIC SIGN ALAYHE ASSALAM~~
~~0612 ARABIC SIGN RAHMATULLAH ALAYHE~~
~~0613 ARABIC SIGN RADI ALLAHOU ANHU~~
~~0614 ARABIC SIGN TAKHALLUS~~
~~0615 ARABIC SMALL HIGH TAH~~
~~0616 ARABIC SMALL HIGH LIGATURE ALEF WITH LAM WITH YEH~~
~~0617 ARABIC SMALL HIGH ZAIN~~
~~0618 SMALL FATHA~~
~~0619 ARABIC SMALL DAMMA~~
~~061A ARABIC SMALL KASRA~~
~~064B ARABIC FATHATAN~~
~~064C ARABIC DAMMATAN~~
~~064D ARABIC KASRATAN~~
~~064E ARABIC FATHA~~
~~064F ARABIC DAMMA~~
~~0650 ARABIC KASRA~~
~~0651 ARABIC SHADDA~~
~~0652 ARABIC SUKUN~~
~~0653 ARABIC MADDAH ABOVE~~
~~0654 ARABIC HAMZA ABOVE~~
~~0655 ARABIC HAMZA BELOW~~
~~0656 ARABIC SUBSCRIPT ALEF~~
~~0657 ARABIC INVERTED DAMMA~~
~~0658 ARABIC NOON GHUNNA~~
~~0659 ARABIC ZWARAKAY~~
~~065A ARABIC VOWEL SIGN SMALL V ABOVE~~
~~065B ARABIC VOWEL SIGN INVERTED SMALL V ABOVE~~
~~065C ARABIC VOWEL SIGN DOT BELOW~~
~~065D ARABIC REVERSED DAMMA~~
~~065E ARABIC FATHA WITH TWO DOTS~~
~~0670 ARABIC LETTER SUPERSCRIPT ALEF~~
~~06D7 ARABIC SMALL HIGH LIGATURE QAF WITH LAM WITH ALEF MAKSURA~~
~~06D8 ARABIC SMALL HIGH MEEM INITIAL FORM~~
~~06D9 ARABIC SMALL HIGH LAM ALEF~~
~~06DA ARABIC SMALL HIGH JEEM~~
~~06DB ARABIC SMALL HIGH THREE DOTS~~
~~06DC ARABIC SMALL HIGH SEEN~~
~~06DE ARABIC START OF RUB EL HIZB~~
~~06DF ARABIC SMALL HIGH ROUNDED ZERO~~
~~06E0 ARABIC SMALL HIGH UPRIGHT RECTANGULAR ZERO~~
~~06E1 ARABIC SMALL HIGH DOTLESS HEAD OF KHAH~~
~~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~~
~~06ED ARABIC SMALL LOW MEEM~~
~~0711 SYRIAC LETTER SUPERSCRIPT ALAPH~~
~~0730 SYRIAC PTHAHA ABOVE~~

~~0731 SYRIAC PTHAHA BELOW~~
~~0732 SYRIAC PTHAHA DOTTED~~
~~0733 SYRIAC ZQAPHA ABOVE~~
~~0734 SYRIAC ZQAPHA BELOW~~
~~0735 SYRIAC ZQAPHA DOTTED~~
~~0736 SYRIAC RBASA ABOVE~~
~~0737 SYRIAC RBASA BELOW~~
~~0738 SYRIAC DOTTED ZLAMA HORIZONTAL~~
~~0739 SYRIAC DOTTED ZLAMA ANGULAR~~
~~073A SYRIAC HBASA ABOVE~~
~~073B SYRIAC HBASA BELOW~~
~~073C SYRIAC HBASA-ESASA DOTTED~~
~~073D SYRIAC ESASA ABOVE~~
~~073E SYRIAC ESASA BELOW~~
~~073F SYRIAC RWAHA~~
~~0740 SYRIAC FEMININE DOT~~
~~0741 SYRIAC QUSHSHAYA~~
~~0742 SYRIAC RUKKAKHA~~
~~0743 SYRIAC TWO VERTICAL DOTS ABOVE~~
~~0744 SYRIAC TWO VERTICAL DOTS BELOW~~
~~0745 SYRIAC THREE DOTS ABOVE~~
~~0746 SYRIAC THREE DOTS BELOW~~
~~0747 SYRIAC OBLIQUE LINE ABOVE~~
~~0748 SYRIAC OBLIQUE LINE BELOW~~
~~0749 SYRIAC MUSIC~~
~~074A SYRIAC BARREKH~~
~~07A6 THAANA ABAFILI~~
~~07A7 THAANA AABAAFILI~~
~~07A8 THAANA IBIFILI~~
~~07A9 THAANA EEBEEFILI~~
~~07AA THAANA UBUFILI~~
~~07AB THAANA OOBOOFILI~~
~~07AC THAANA EBEFILI~~
~~07AD THAANA EYBEYFILI~~
~~07AE THAANA OBOFILI~~
~~07AF THAANA OABOAFILI~~
~~07B0 THAANA SUKUN~~
~~07EB NKO COMBINING SHORT HIGH TONE~~
~~07EC NKO COMBINING SHORT LOW TONE~~
~~07ED NKO COMBINING SHORT RISING TONE~~
~~07EE NKO COMBINING LONG DESCENDING TONE~~
~~07EF NKO COMBINING LONG HIGH TONE~~
~~07F0 NKO COMBINING LONG LOW TONE~~
~~07F1 NKO COMBINING LONG RISING TONE~~
~~07F2 NKO COMBINING NASALIZATION MARK~~
~~07F3 NKO COMBINING DOUBLE DOT ABOVE~~
~~0901 DEVANAGARI SIGN CANDRABINDU~~
~~0902 DEVANAGARI SIGN ANUSVARA~~
~~0903 DEVANAGARI SIGN VISARGA~~
~~093C DEVANAGARI SIGN NUKTA~~
~~093E DEVANAGARI VOWEL SIGN AA~~
~~093F DEVANAGARI VOWEL SIGN I~~
~~0940 DEVANAGARI VOWEL SIGN II~~
~~0941 DEVANAGARI VOWEL SIGN U~~
~~0942 DEVANAGARI VOWEL SIGN UU~~
~~0943 DEVANAGARI VOWEL SIGN VOCALIC R~~
~~0944 DEVANAGARI VOWEL SIGN VOCALIC RR~~
~~0945 DEVANAGARI VOWEL SIGN CANDRA E~~
~~0946 DEVANAGARI VOWEL SIGN SHORT E~~
~~0947 DEVANAGARI VOWEL SIGN E~~
~~0948 DEVANAGARI VOWEL SIGN AI~~
~~0949 DEVANAGARI VOWEL SIGN CANDRA O~~
~~094A DEVANAGARI VOWEL SIGN SHORT O~~
~~094B DEVANAGARI VOWEL SIGN O~~
~~094C DEVANAGARI VOWEL SIGN AU~~
~~094D DEVANAGARI SIGN VIRAMA~~

~~0951 DEVANAGARI STRESS SIGN UDATTA~~
~~0952 DEVANAGARI STRESS SIGN ANUDATTA~~
~~0953 DEVANAGARI GRAVE ACCENT~~
~~0954 DEVANAGARI ACUTE ACCENT~~
~~0962 DEVANAGARI VOWEL SIGN VOCALIC L~~
~~0963 DEVANAGARI VOWEL SIGN VOCALIC LL~~
~~0981 BENGALI SIGN CANDRABINDU~~
~~0982 BENGALI SIGN ANUSVARA~~
~~0983 BENGALI SIGN VISARGA~~
~~09BC BENGALI SIGN NUKTA~~
~~09BE BENGALI VOWEL SIGN AA~~
~~09BF BENGALI VOWEL SIGN I~~
~~09C0 BENGALI VOWEL SIGN II~~
~~09C1 BENGALI VOWEL SIGN U~~
~~09C2 BENGALI VOWEL SIGN UU~~
~~09C3 BENGALI VOWEL SIGN VOCALIC R~~
~~09C4 BENGALI VOWEL SIGN VOCALIC RR~~
~~09C7 BENGALI VOWEL SIGN E~~
~~09C8 BENGALI VOWEL SIGN AI~~
~~09CB BENGALI VOWEL SIGN O~~
~~09CC BENGALI VOWEL SIGN AU~~
~~09CD BENGALI SIGN VIRAMA~~
~~09D7 BENGALI AU LENGTH MARK~~
~~09E2 BENGALI VOWEL SIGN VOCALIC L~~
~~09E3 BENGALI VOWEL SIGN VOCALIC LL~~
~~0A01 GURMUKHI SIGN ADAK BINDI~~
~~0A02 GURMUKHI SIGN BINDI~~
~~0A03 GURMUKHI SIGN VISARGA~~
~~0A3C GURMUKHI SIGN NUKTA~~
~~0A3E GURMUKHI VOWEL SIGN AA~~
~~0A3F GURMUKHI VOWEL SIGN I~~
~~0A40 GURMUKHI VOWEL SIGN II~~
~~0A41 GURMUKHI VOWEL SIGN U~~
~~0A42 GURMUKHI VOWEL SIGN UU~~
~~0A47 GURMUKHI VOWEL SIGN EE~~
~~0A48 GURMUKHI VOWEL SIGN AI~~
~~0A4B GURMUKHI VOWEL SIGN OO~~
~~0A4C GURMUKHI VOWEL SIGN AU~~
~~0A4D GURMUKHI SIGN VIRAMA~~
~~0A51 GURMUKHI SIGN UDAAT~~
~~0A70 GURMUKHI TIPPI~~
~~0A71 GURMUKHI ADDAK~~
~~0A75 GURMUKHI SIGN YAKASH~~
~~0A81 GUJARATI SIGN CANDRABINDU~~
~~0A82 GUJARATI SIGN ANUSVARA~~
~~0A83 GUJARATI SIGN VISARGA~~
~~0ABC GUJARATI SIGN NUKTA~~
~~0ABE GUJARATI VOWEL SIGN AA~~
~~0ABF GUJARATI VOWEL SIGN I~~
~~0AC0 GUJARATI VOWEL SIGN II~~
~~0AC1 GUJARATI VOWEL SIGN U~~
~~0AC2 GUJARATI VOWEL SIGN UU~~
~~0AC3 GUJARATI VOWEL SIGN VOCALIC R~~
~~0AC4 GUJARATI VOWEL SIGN VOCALIC RR~~
~~0AC5 GUJARATI VOWEL SIGN CANDRA E~~
~~0AC7 GUJARATI VOWEL SIGN E~~
~~0AC8 GUJARATI VOWEL SIGN AI~~
~~0AC9 GUJARATI VOWEL SIGN CANDRA O~~
~~0ACB GUJARATI VOWEL SIGN O~~
~~0ACC GUJARATI VOWEL SIGN AU~~
~~0ACD GUJARATI SIGN VIRAMA~~
~~0AE2 GUJARATI VOWEL SIGN VOCALIC L~~
~~0AE3 GUJARATI VOWEL SIGN VOCALIC LL~~
~~0B01 ORIYA SIGN CANDRABINDU~~
~~0B02 ORIYA SIGN ANUSVARA~~
~~0B03 ORIYA SIGN VISARGA~~

~~0B3C ORIYA SIGN NUKTA~~
~~0B3E ORIYA VOWEL SIGN AA~~
~~0B3F ORIYA VOWEL SIGN I~~
~~0B40 ORIYA VOWEL SIGN II~~
~~0B41 ORIYA VOWEL SIGN U~~
~~0B42 ORIYA VOWEL SIGN UU~~
~~0B43 ORIYA VOWEL SIGN VOCALIC R~~
~~0B44 ORIYA VOWEL SIGN VOCALIC RR~~
~~0B47 ORIYA VOWEL SIGN E~~
~~0B48 ORIYA VOWEL SIGN AI~~
~~0B4B ORIYA VOWEL SIGN O~~
~~0B4C ORIYA VOWEL SIGN AU~~
~~0B4D ORIYA SIGN VIRAMA~~
~~0B56 ORIYA AI LENGTH MARK~~
~~0B57 ORIYA AU LENGTH MARK~~
~~0B62 ORIYA VOWEL SIGN VOCALIC L~~
~~0B63 ORIYA VOWEL SIGN VOCALIC LL~~
~~0B82 TAMIL SIGN ANUSVARA~~
~~0BBE TAMIL VOWEL SIGN AA~~
~~0BBF TAMIL VOWEL SIGN I~~
~~0BC0 TAMIL VOWEL SIGN II~~
~~0BC1 TAMIL VOWEL SIGN U~~
~~0BC2 TAMIL VOWEL SIGN UU~~
~~0BC6 TAMIL VOWEL SIGN E~~
~~0BC7 TAMIL VOWEL SIGN EE~~
~~0BC8 TAMIL VOWEL SIGN AI~~
~~0BCA TAMIL VOWEL SIGN O~~
~~0BCB TAMIL VOWEL SIGN OO~~
~~0BCC TAMIL VOWEL SIGN AU~~
~~0BCD TAMIL SIGN VIRAMA~~
~~0BD7 TAMIL AU LENGTH MARK~~
~~0C01 TELUGU SIGN CANDRABINDU~~
~~0C02 TELUGU SIGN ANUSVARA~~
~~0C03 TELUGU SIGN VISARGA~~
~~0C3E TELUGU VOWEL SIGN AA~~
~~0C3F TELUGU VOWEL SIGN I~~
~~0C40 TELUGU VOWEL SIGN II~~
~~0C41 TELUGU VOWEL SIGN U~~
~~0C42 TELUGU VOWEL SIGN UU~~
~~0C43 TELUGU VOWEL SIGN VOCALIC R~~
~~0C44 TELUGU VOWEL SIGN VOCALIC RR~~
~~0C46 TELUGU VOWEL SIGN E~~
~~0C47 TELUGU VOWEL SIGN EE~~
~~0C48 TELUGU VOWEL SIGN AI~~
~~0C4A TELUGU VOWEL SIGN O~~
~~0C4B TELUGU VOWEL SIGN OO~~
~~0C4C TELUGU VOWEL SIGN AU~~
~~0C4D TELUGU SIGN VIRAMA~~
~~0C55 TELUGU LENGTH MARK~~
~~0C56 TELUGU AI LENGTH MARK~~
~~0C62 TELUGU VOWEL SIGN VOCALIC L~~
~~0C63 TELUGU VOWEL SIGN VOCALIC LL~~
~~0C82 KANNADA SIGN ANUSVARA~~
~~0C83 KANNADA SIGN VISARGA~~
~~0CBC KANNADA SIGN NUKTA~~
~~0CBE KANNADA VOWEL SIGN AA~~
~~0CBF KANNADA VOWEL SIGN I~~
~~0CC0 KANNADA VOWEL SIGN II~~
~~0CC1 KANNADA VOWEL SIGN U~~
~~0CC2 KANNADA VOWEL SIGN UU~~
~~0CC3 KANNADA VOWEL SIGN VOCALIC R~~
~~0CC4 KANNADA VOWEL SIGN VOCALIC RR~~
~~0CC6 KANNADA VOWEL SIGN E~~
~~0CC7 KANNADA VOWEL SIGN EE~~
~~0CC8 KANNADA VOWEL SIGN AI~~
~~0CCA KANNADA VOWEL SIGN O~~

~~0CCB KANNADA VOWEL SIGN OO~~
~~0CCC KANNADA VOWEL SIGN AU~~
~~0CCD KANNADA SIGN VIRAMA~~
~~0CD5 KANNADA LENGTH MARK~~
~~0CD6 KANNADA AI LENGTH MARK~~
~~0CE2 KANNADA VOWEL SIGN VOCALIC L~~
~~0CE3 KANNADA VOWEL SIGN VOCALIC LL~~
~~0D02 MALAYALAM SIGN ANUSVARA~~
~~0D03 MALAYALAM SIGN VISARGA~~
~~0D3E MALAYALAM VOWEL SIGN AA~~
~~0D3F MALAYALAM VOWEL SIGN I~~
~~0D40 MALAYALAM VOWEL SIGN II~~
~~0D41 MALAYALAM VOWEL SIGN U~~
~~0D42 MALAYALAM VOWEL SIGN UU~~
~~0D43 MALAYALAM VOWEL SIGN VOCALIC R~~
~~0D44 MALAYALAM VOWEL SIGN VOCALIC RR~~
~~0D46 MALAYALAM VOWEL SIGN E~~
~~0D47 MALAYALAM VOWEL SIGN EE~~
~~0D48 MALAYALAM VOWEL SIGN AI~~
~~0D4A MALAYALAM VOWEL SIGN O~~
~~0D4B MALAYALAM VOWEL SIGN OO~~
~~0D4C MALAYALAM VOWEL SIGN AU~~
~~0D4D MALAYALAM SIGN VIRAMA~~
~~0D57 MALAYALAM AU LENGTH MARK~~
~~0D62 MALAYALAM VOWEL SIGN VOCALIC L~~
~~0D63 MALAYALAM VOWEL SIGN VOCALIC LL~~
~~0D82 SINHALA SIGN ANUSVARAYA~~
~~0D83 SINHALA SIGN VISARGAYA~~
~~0DCA SINHALA SIGN AL-LAKUNA~~
~~0DCF SINHALA VOWEL SIGN AELA-PILLA~~
~~0DD0 SINHALA VOWEL SIGN KETTI AEDA-PILLA~~
~~0DD1 SINHALA VOWEL SIGN DIGA AEDA-PILLA~~
~~0DD2 SINHALA VOWEL SIGN KETTI IS-PILLA~~
~~0DD3 SINHALA VOWEL SIGN DIGA IS-PILLA~~
~~0DD4 SINHALA VOWEL SIGN KETTI PAA-PILLA~~
~~0DD6 SINHALA VOWEL SIGN DIGA PAA-PILLA~~
~~0DD8 SINHALA VOWEL SIGN GAETTA-PILLA~~
~~0DD9 SINHALA VOWEL SIGN KOMBUVA~~
~~0DDA SINHALA VOWEL SIGN DIGA KOMBUVA~~
~~0DDB SINHALA VOWEL SIGN KOMBU DEKA~~
~~0DDC SINHALA VOWEL SIGN KOMBUVA HAA AELA-PILLA~~
~~0DDD SINHALA VOWEL SIGN KOMBUVA HAA DIGA AELA-PILLA~~
~~0DDE SINHALA VOWEL SIGN KOMBUVA HAA GAYANUKITTA~~
~~0DDF SINHALA VOWEL SIGN GAYANUKITTA~~
~~0DF2 SINHALA VOWEL SIGN DIGA GAETTA-PILLA~~
~~0DF3 SINHALA VOWEL SIGN DIGA GAYANUKITTA~~
~~0E31 THAI CHARACTER MAI HAN-AKAT~~
~~0E34 THAI CHARACTER SARA I~~
~~0E35 THAI CHARACTER SARA II~~
~~0E36 THAI CHARACTER SARA UE~~
~~0E37 THAI CHARACTER SARA UEE~~
~~0E38 THAI CHARACTER SARA U~~
~~0E39 THAI CHARACTER SARA UU~~
~~0E3A THAI CHARACTER PHINTHU~~
~~0E47 THAI CHARACTER MAITAIKHU~~
~~0E48 THAI CHARACTER MAI EK~~
~~0E49 THAI CHARACTER MAI THO~~
~~0E4A THAI CHARACTER MAI TRI~~
~~0E4B THAI CHARACTER MAI CHATTAWA~~
~~0E4C THAI CHARACTER THANTHAKHAT~~
~~0E4D THAI CHARACTER NIKHAHIT~~
~~0E4E THAI CHARACTER YAMAKKAN~~
~~0EB1 LAO VOWEL SIGN MAI KAN~~
~~0EB4 LAO VOWEL SIGN I~~
~~0EB5 LAO VOWEL SIGN II~~
~~0EB6 LAO VOWEL SIGN Y~~

~~0EB7 LAO VOWEL SIGN YY~~
~~0EB8 LAO VOWEL SIGN U~~
~~0EB9 LAO VOWEL SIGN UU~~
~~0EBB LAO VOWEL SIGN MAI KON~~
~~0EBC LAO SEMIVOWEL SIGN LO~~
~~0EC8 LAO TONE MAI EK~~
~~0EC9 LAO TONE MAI THO~~
~~0ECA LAO TONE MAI TI~~
~~0ECB LAO TONE MAI CATAWA~~
~~0ECC LAO CANCELLATION MARK~~
~~0ECD LAO NIGGAHITA~~
~~0F18 TIBETAN ASTROLOGICAL SIGN -KHYUD PA~~
~~0F19 TIBETAN ASTROLOGICAL SIGN SDONG TSHUGS~~
~~0F35 TIBETAN MARK NGAS BZUNG NYI ZLA~~
~~0F37 TIBETAN MARK NGAS BZUNG SGOR RTAGS~~
~~0F39 TIBETAN MARK TSA -PHRU~~
~~0F3E TIBETAN SIGN YAR TSHES~~
~~0F3F TIBETAN SIGN MAR TSHES~~
~~0F71 TIBETAN VOWEL SIGN AA~~
~~0F72 TIBETAN VOWEL SIGN I~~
~~0F73 TIBETAN VOWEL SIGN II~~
~~0F74 TIBETAN VOWEL SIGN U~~
~~0F75 TIBETAN VOWEL SIGN UU~~
~~0F76 TIBETAN VOWEL SIGN VOCALIC R~~
~~0F77 TIBETAN VOWEL SIGN VOCALIC RR~~
~~0F78 TIBETAN VOWEL SIGN VOCALIC L~~
~~0F79 TIBETAN VOWEL SIGN VOCALIC LL~~
~~0F7A TIBETAN VOWEL SIGN E~~
~~0F7B TIBETAN VOWEL SIGN EE~~
~~0F7C TIBETAN VOWEL SIGN O~~
~~0F7D TIBETAN VOWEL SIGN OO~~
~~0F7E TIBETAN SIGN RJES SU NGA RO~~
~~0F7F TIBETAN SIGN RNAM BCAD~~
~~0F80 TIBETAN VOWEL SIGN REVERSED I~~
~~0F81 TIBETAN VOWEL SIGN REVERSED II~~
~~0F82 TIBETAN SIGN NYI ZLA NAA DA~~
~~0F83 TIBETAN SIGN SNA LDAN~~
~~0F84 TIBETAN MARK HALANTA~~
~~0F86 TIBETAN MARK LCI RTAGS~~
~~0F87 TIBETAN MARK YANG RTAGS~~
~~0F90 TIBETAN SUBJOINED LETTER KA~~
~~0F91 TIBETAN SUBJOINED LETTER KHA~~
~~0F92 TIBETAN SUBJOINED LETTER GA~~
~~0F93 TIBETAN SUBJOINED LETTER GHA~~
~~0F94 TIBETAN SUBJOINED LETTER NGA~~
~~0F95 TIBETAN SUBJOINED LETTER CA~~
~~0F96 TIBETAN SUBJOINED LETTER CHA~~
~~0F97 TIBETAN SUBJOINED LETTER JA~~
~~0F99 TIBETAN SUBJOINED LETTER NYA~~
~~0F9A TIBETAN SUBJOINED LETTER TTA~~
~~0F9B TIBETAN SUBJOINED LETTER TTHA~~
~~0F9C TIBETAN SUBJOINED LETTER DDA~~
~~0F9D TIBETAN SUBJOINED LETTER DDHA~~
~~0F9E TIBETAN SUBJOINED LETTER NNA~~
~~0F9F TIBETAN SUBJOINED LETTER TA~~
~~0FA0 TIBETAN SUBJOINED LETTER THA~~
~~0FA1 TIBETAN SUBJOINED LETTER DA~~
~~0FA2 TIBETAN SUBJOINED LETTER DHA~~
~~0FA3 TIBETAN SUBJOINED LETTER NA~~
~~0FA4 TIBETAN SUBJOINED LETTER PA~~
~~0FA5 TIBETAN SUBJOINED LETTER PHA~~
~~0FA6 TIBETAN SUBJOINED LETTER BA~~
~~0FA7 TIBETAN SUBJOINED LETTER BHA~~
~~0FA8 TIBETAN SUBJOINED LETTER MA~~
~~0FA9 TIBETAN SUBJOINED LETTER TSA~~
~~0FAA TIBETAN SUBJOINED LETTER TSHA~~

~~0FAB TIBETAN SUBJOINED LETTER DZA~~
~~0FAC TIBETAN SUBJOINED LETTER DZHA~~
~~0FAD TIBETAN SUBJOINED LETTER WA~~
~~0FAE TIBETAN SUBJOINED LETTER ZHA~~
~~0FAF TIBETAN SUBJOINED LETTER ZA~~
~~0FB0 TIBETAN SUBJOINED LETTER -A~~
~~0FB1 TIBETAN SUBJOINED LETTER YA~~
~~0FB2 TIBETAN SUBJOINED LETTER RA~~
~~0FB3 TIBETAN SUBJOINED LETTER LA~~
~~0FB4 TIBETAN SUBJOINED LETTER SHA~~
~~0FB5 TIBETAN SUBJOINED LETTER SSA~~
~~0FB6 TIBETAN SUBJOINED LETTER SA~~
~~0FB7 TIBETAN SUBJOINED LETTER HA~~
~~0FB8 TIBETAN SUBJOINED LETTER A~~
~~0FB9 TIBETAN SUBJOINED LETTER KSSA~~
~~0FBA TIBETAN SUBJOINED LETTER FIXED-FORM WA~~
~~0FBB TIBETAN SUBJOINED LETTER FIXED-FORM YA~~
~~0FBC TIBETAN SUBJOINED LETTER FIXED-FORM RA~~
~~0FC6 TIBETAN SYMBOL PADMA GDAN~~
~~102B MYANMAR VOWEL SIGN TALL AA~~
~~102C MYANMAR VOWEL SIGN AA~~
~~102D MYANMAR VOWEL SIGN I~~
~~102E MYANMAR VOWEL SIGN II~~
~~102F MYANMAR VOWEL SIGN U~~
~~1030 MYANMAR VOWEL SIGN UU~~
~~1031 MYANMAR VOWEL SIGN E~~
~~1032 MYANMAR VOWEL SIGN AI~~
~~1033 MYANMAR VOWEL SIGN MON II~~
~~1034 MYANMAR VOWEL SIGN MON O~~
~~1035 MYANMAR VOWEL SIGN E ABOVE~~
~~1036 MYANMAR SIGN ANUSVARA~~
~~1037 MYANMAR SIGN DOT BELOW~~
~~1038 MYANMAR SIGN VISARGA~~
~~1039 MYANMAR SIGN VIRAMA~~
~~103A MYANMAR SIGN ASAT~~
~~103B MYANMAR CONSONANT SIGN MEDIAL YA~~
~~103C MYANMAR CONSONANT SIGN MEDIAL RA~~
~~103D MYANMAR CONSONANT SIGN MEDIAL WA~~
~~103E MYANMAR CONSONANT SIGN MEDIAL HA~~
~~1056 MYANMAR VOWEL SIGN VOCALIC R~~
~~1057 MYANMAR VOWEL SIGN VOCALIC RR~~
~~1058 MYANMAR VOWEL SIGN VOCALIC L~~
~~1059 MYANMAR VOWEL SIGN VOCALIC LL~~
~~105E MYANMAR CONSONANT SIGN MON MEDIAL NA~~
~~105F MYANMAR CONSONANT SIGN MON MEDIAL MA~~
~~1060 MYANMAR CONSONANT SIGN MON MEDIAL LA~~
~~1062 MYANMAR LETTER SGAW KAREN EU~~
~~1063 MYANMAR TONE MARK SGAW KAREN HATHI~~
~~1064 MYANMAR TONE MARK SGAW KAREN KE PHO~~
~~1067 MYANMAR VOWEL SIGN WESTERN PWO KAREN EU~~
~~1068 MYANMAR VOWEL SIGN WESTERN PWO KAREN UE~~
~~1069 MYANMAR SIGN WESTERN PWO KAREN TONE 1~~
~~106A MYANMAR SIGN WESTERN PWO KAREN TONE 2~~
~~106B MYANMAR SIGN WESTERN PWO KAREN TONE 3~~
~~106C MYANMAR SIGN WESTERN PWO KAREN TONE 4~~
~~106D MYANMAR SIGN WESTERN PWO KAREN TONE 5~~
~~1071 MYANMAR VOWEL SIGN GEBA KAREN I~~
~~1072 MYANMAR VOWEL SIGN KAYAH OE~~
~~1073 MYANMAR VOWEL SIGN KAYAH U~~
~~1074 MYANMAR VOWEL SIGN KAYAH EE~~
~~107E MYANMAR CONSONANT SIGN SHAN MEDIAL WA~~

~~107F MYANMAR VOWEL SIGN SHAN AA~~
~~1080 MYANMAR VOWEL SIGN SHAN E~~
~~1081 MYANMAR VOWEL SIGN SHAN E ABOVE~~
~~1082 MYANMAR VOWEL SIGN SHAN FINAL Y~~
~~1083 MYANMAR SIGN SHAN TONE-2~~
~~1084 MYANMAR SIGN SHAN TONE-3~~
~~1085 MYANMAR SIGN SHAN COUNCIL TONE-4~~
~~1086 MYANMAR SIGN SHAN TONE-5~~
~~1087 MYANMAR SIGN SHAN TONE-6~~
~~1088 MYANMAR SIGN SHAN COUNCIL EMPHATIC TONE~~
~~108A MYANMAR SIGN RUMAI PALAUNG TONE-5~~
~~135F ETHIOPIC COMBINING GEMINATION MARK~~
~~1712 TAGALOG VOWEL SIGN I~~
~~1713 TAGALOG VOWEL SIGN U~~
~~1714 TAGALOG VIRAMA~~
~~1732 HANUNOO VOWEL SIGN I~~
~~1733 HANUNOO VOWEL SIGN U~~
~~1734 HANUNOO PAMUDPOD~~
~~1752 BUHID VOWEL SIGN I~~
~~1753 BUHID VOWEL SIGN U~~
~~1772 TAGBANWA VOWEL SIGN I~~
~~1773 TAGBANWA VOWEL SIGN U~~
~~17B6 KHMER VOWEL SIGN AA~~
~~17B7 KHMER VOWEL SIGN I~~
~~17B8 KHMER VOWEL SIGN II~~
~~17B9 KHMER VOWEL SIGN Y~~
~~17BA KHMER VOWEL SIGN YY~~
~~17BB KHMER VOWEL SIGN U~~
~~17BC KHMER VOWEL SIGN UU~~
~~17BD KHMER VOWEL SIGN UA~~
~~17BE KHMER VOWEL SIGN OE~~
~~17BF KHMER VOWEL SIGN YA~~
~~17C0 KHMER VOWEL SIGN IE~~
~~17C1 KHMER VOWEL SIGN E~~
~~17C2 KHMER VOWEL SIGN AE~~
~~17C3 KHMER VOWEL SIGN AI~~
~~17C4 KHMER VOWEL SIGN OO~~
~~17C5 KHMER VOWEL SIGN AU~~
~~17C6 KHMER SIGN NIKAHIT~~
~~17C7 KHMER SIGN REAHMUK~~
~~17C8 KHMER SIGN YUUKALEAPINTU~~
~~17C9 KHMER SIGN MUUSIKATOAN~~
~~17CA KHMER SIGN TRIISAP~~
~~17CB KHMER SIGN BANTOC~~
~~17CC KHMER SIGN ROBAT~~
~~17CD KHMER SIGN TOANDAKHIAT~~
~~17CE KHMER SIGN KAKABAT~~
~~17CF KHMER SIGN AHSDA~~
~~17D0 KHMER SIGN SAMYOK SANNYA~~
~~17D1 KHMER SIGN VIRIAM~~
~~17D2 KHMER SIGN COENG~~
~~17D3 KHMER SIGN BATHAMASAT~~
~~17DD KHMER SIGN ATTHACAN~~
~~180B MONGOLIAN FREE VARIATION SELECTOR ONE~~
~~180C MONGOLIAN FREE VARIATION SELECTOR TWO~~
~~180D MONGOLIAN FREE VARIATION SELECTOR THREE~~
~~18A9 MONGOLIAN LETTER ALI GALI DAGALGA~~
~~1920 LIMBU VOWEL SIGN A~~
~~1921 LIMBU VOWEL SIGN I~~
~~1922 LIMBU VOWEL SIGN U~~
~~1923 LIMBU VOWEL SIGN EE~~
~~1924 LIMBU VOWEL SIGN AI~~
~~1925 LIMBU VOWEL SIGN OO~~
~~1926 LIMBU VOWEL SIGN AU~~
~~1927 LIMBU VOWEL SIGN E~~
~~1928 LIMBU VOWEL SIGN O~~

~~1929 LIMBU SUBJOINED LETTER YA~~
~~192A LIMBU SUBJOINED LETTER RA~~
~~192B LIMBU SUBJOINED LETTER WA~~
~~1930 LIMBU SMALL LETTER KA~~
~~1931 LIMBU SMALL LETTER NGA~~
~~1932 LIMBU SMALL LETTER ANUSVARA~~
~~1933 LIMBU SMALL LETTER TA~~
~~1934 LIMBU SMALL LETTER NA~~
~~1935 LIMBU SMALL LETTER PA~~
~~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~~
~~19B0 NEW TAI LUE VOWEL SIGN VOWEL SHORTENER~~
~~19B1 NEW TAI LUE VOWEL SIGN AA~~
~~19B2 NEW TAI LUE VOWEL SIGN II~~
~~19B3 NEW TAI LUE VOWEL SIGN U~~
~~19B4 NEW TAI LUE VOWEL SIGN UU~~
~~19B5 NEW TAI LUE VOWEL SIGN EE~~
~~19B6 NEW TAI LUE VOWEL SIGN AE~~
~~19B7 NEW TAI LUE VOWEL SIGN O~~
~~19B8 NEW TAI LUE VOWEL SIGN OA~~
~~19B9 NEW TAI LUE VOWEL SIGN UE~~
~~19BA NEW TAI LUE VOWEL SIGN AY~~
~~19BB NEW TAI LUE VOWEL SIGN AAY~~
~~19BC NEW TAI LUE VOWEL SIGN UY~~
~~19BD NEW TAI LUE VOWEL SIGN OY~~
~~19BE NEW TAI LUE VOWEL SIGN OAY~~
~~19BF NEW TAI LUE VOWEL SIGN UEY~~
~~19C0 NEW TAI LUE VOWEL SIGN IY~~
~~19C8 NEW TAI LUE TONE MARK-1~~
~~19C9 NEW TAI LUE TONE MARK-2~~
~~1A17 BUGINESE VOWEL SIGN I~~
~~1A18 BUGINESE VOWEL SIGN U~~
~~1A19 BUGINESE VOWEL SIGN E~~
~~1A1A BUGINESE VOWEL SIGN O~~
~~1A1B BUGINESE VOWEL SIGN AE~~
~~1A58 LANNA CONSONANT SIGN MEDIAL RA~~
~~1A59 LANNA CONSONANT SIGN MEDIAL LA~~
~~1A5A LANNA SIGN MAI KANG LAI~~
~~1A5B LANNA SIGN KHUEN MAI KANG LAI~~
~~1A5C LANNA CONSONANT SIGN FINAL NGA~~
~~1A5D LANNA CONSONANT SIGN LOW PA~~
~~1A5E LANNA CONSONANT SIGN HIGH RATHA OR LOW PA~~
~~1A60 LANNA SIGN SAKOT~~
~~1A61 LANNA VOWEL SIGN A~~
~~1A62 LANNA VOWEL SIGN MAI SAT~~
~~1A63 LANNA VOWEL SIGN AA~~
~~1A64 LANNA VOWEL SIGN TALL AA~~
~~1A65 LANNA VOWEL SIGN I~~
~~1A66 LANNA VOWEL SIGN II~~
~~1A67 LANNA VOWEL SIGN UE~~
~~1A68 LANNA VOWEL SIGN UUE~~
~~1A69 LANNA VOWEL SIGN U~~
~~1A6A LANNA VOWEL SIGN UU~~
~~1A6B LANNA VOWEL SIGN O~~
~~1A6C LANNA VOWEL SIGN OA BELOW~~
~~1A6D LANNA VOWEL SIGN OY~~
~~1A6E LANNA VOWEL SIGN E~~
~~1A6F LANNA VOWEL SIGN AE~~
~~1A70 LANNA VOWEL SIGN OO~~
~~1A71 LANNA VOWEL SIGN AI~~
~~1A72 LANNA VOWEL SIGN THAM AI~~
~~1A73 LANNA VOWEL SIGN OA ABOVE~~

~~1A74 LANNA SIGN MAI KANG~~
~~1A75 LANNA SIGN TONE-1~~
~~1A76 LANNA SIGN TONE-2~~
~~1A77 LANNA SIGN KHUEN TONE-3~~
~~1A78 LANNA SIGN KHUEN TONE-4~~
~~1A79 LANNA SIGN KHUEN TONE-5~~
~~1A7A LANNA SIGN RA HAAM~~
~~1A7B LANNA SIGN MAI SAM~~
~~1A7F LANNA COMBINING CRYPTOGRAMMIC DOT~~
~~1B00 BALINESE SIGN ULU RICEM~~
~~1B01 BALINESE SIGN ULU CANDRA~~
~~1B02 BALINESE SIGN CECEK~~
~~1B03 BALINESE SIGN SURANG~~
~~1B04 BALINESE SIGN BISAH~~
~~1B34 BALINESE SIGN REREKAN~~
~~1B35 BALINESE VOWEL SIGN TEDUNG~~
~~1B36 BALINESE VOWEL SIGN ULU~~
~~1B37 BALINESE VOWEL SIGN ULU SARI~~
~~1B38 BALINESE VOWEL SIGN SUKU~~
~~1B39 BALINESE VOWEL SIGN SUKU ILUT~~
~~1B3A BALINESE VOWEL SIGN RA REPA~~
~~1B3B BALINESE VOWEL SIGN RA REPA TEDUNG~~
~~1B3C BALINESE VOWEL SIGN LA LENGA~~
~~1B3D BALINESE VOWEL SIGN LA LENGA TEDUNG~~
~~1B3E BALINESE VOWEL SIGN TALING~~
~~1B3F BALINESE VOWEL SIGN TALING REPA~~
~~1B40 BALINESE VOWEL SIGN TALING TEDUNG~~
~~1B41 BALINESE VOWEL SIGN TALING REPA TEDUNG~~
~~1B42 BALINESE VOWEL SIGN PEPET~~
~~1B43 BALINESE VOWEL SIGN PEPET TEDUNG~~
~~1B44 BALINESE ADEG ADEG~~
~~1B6B BALINESE MUSICAL SYMBOL COMBINING TEGEH~~
~~1B6C BALINESE MUSICAL SYMBOL COMBINING ENDEP~~
~~1B6D BALINESE MUSICAL SYMBOL COMBINING KEMPUL~~
~~1B6E BALINESE MUSICAL SYMBOL COMBINING KEMPLI~~
~~1B6F BALINESE MUSICAL SYMBOL COMBINING JEGOGAN~~
~~1B70 BALINESE MUSICAL SYMBOL COMBINING KEMPUL WITH JEGOGAN~~
~~1B71 BALINESE MUSICAL SYMBOL COMBINING KEMPLI WITH JEGOGAN~~
~~1B72 BALINESE MUSICAL SYMBOL COMBINING BENDE~~
~~1B73 BALINESE MUSICAL SYMBOL COMBINING GONG~~
~~1B80 SUNDANESE SIGN PANYECEK~~
~~1B81 SUNDANESE SIGN PANGLAYAR~~
~~1B82 SUNDANESE SIGN PANGWISAD~~
~~1BA1 SUNDANESE CONSONANT SIGN PAMINGKAL~~
~~1BA2 SUNDANESE CONSONANT SIGN PANYAKRA~~
~~1BA3 SUNDANESE CONSONANT SIGN PANYIKU~~
~~1BA4 SUNDANESE VOWEL SIGN PANGHULU~~
~~1BA5 SUNDANESE VOWEL SIGN PANYUKU~~
~~1BA6 SUNDANESE VOWEL SIGN PANAELAENG~~
~~1BA7 SUNDANESE VOWEL SIGN PANOLONG~~
~~1BA8 SUNDANESE VOWEL SIGN PAMEPET~~
~~1BA9 SUNDANESE VOWEL SIGN PANEULEUNG~~
~~1BAA SUNDANESE SIGN PAMAAEH~~
~~1C24 LEPCHA SUBJOINED LETTER YA~~
~~1C25 LEPCHA SUBJOINED LETTER RA~~
~~1C26 LEPCHA VOWEL SIGN AA~~
~~1C27 LEPCHA VOWEL SIGN I~~
~~1C28 LEPCHA VOWEL SIGN O~~
~~1C29 LEPCHA VOWEL SIGN OO~~
~~1C2A LEPCHA VOWEL SIGN U~~
~~1C2B LEPCHA VOWEL SIGN UU~~
~~1C2C LEPCHA VOWEL SIGN E~~
~~1C2D LEPCHA CONSONANT SIGN K~~
~~1C2E LEPCHA CONSONANT SIGN M~~
~~1C2F LEPCHA CONSONANT SIGN L~~
~~1C30 LEPCHA CONSONANT SIGN N~~

~~1C31 LEPCHA CONSONANT SIGN P~~
~~1C32 LEPCHA CONSONANT SIGN R~~
~~1C33 LEPCHA CONSONANT SIGN T~~
~~1C34 LEPCHA CONSONANT SIGN NYIN-DO~~
~~1C35 LEPCHA CONSONANT SIGN KANG~~
~~1C36 LEPCHA SIGN RAN~~
~~1C37 LEPCHA SIGN NUKTA~~
~~1CA6 MEITEI MAYEK VOWEL SIGN AA~~
~~1CA7 MEITEI MAYEK VOWEL SIGN I~~
~~1CA8 MEITEI MAYEK VOWEL SIGN II~~
~~1CA9 MEITEI MAYEK VOWEL SIGN U~~
~~1CAA MEITEI MAYEK VOWEL SIGN UU~~
~~1CAB MEITEI MAYEK VOWEL SIGN E~~
~~1CAC MEITEI MAYEK VOWEL SIGN EI~~
~~1CAD MEITEI MAYEK VOWEL SIGN AAI~~
~~1CAE MEITEI MAYEK VOWEL SIGN O~~
~~1CAF MEITEI MAYEK VOWEL SIGN OU~~
~~1CB0 MEITEI MAYEK VOWEL SIGN AU~~
~~1CB1 MEITEI MAYEK VOWEL SIGN AAU~~
~~1CB2 MEITEI MAYEK VOWEL SIGN ANUSVARA~~
~~1CB3 MEITEI MAYEK VOWEL SIGN VISARGA~~
~~1CB4 MEITEI MAYEK HEAVY TONE~~
~~1CB5 MEITEI MAYEK KILLER~~
~~1CBF MEITEI MAYEK SIGN VIRAMA~~
~~2CEF COPTIC COMBINING NI ABOVE~~
~~2CF0 COPTIC COMBINING SPIRITUS ASPER~~
~~2CF1 COPTIC COMBINING SPIRITUS LENIS~~
~~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~~
~~A66F COMBINING CYRILLIC VZMET~~
~~A670 COMBINING CYRILLIC TEN MILLIONS SIGN~~
~~A671 COMBINING CYRILLIC HUNDRED MILLIONS SIGN~~
~~A672 COMBINING CYRILLIC THOUSAND MILLIONS SIGN~~
~~A67C COMBINING CYRILLIC KAVYKA~~
~~A67D COMBINING CYRILLIC PAYEROK~~
~~A6F0 BAMUM COMBINING MARK KOQNDON~~
~~A6F1 BAMUM COMBINING MARK TUKWENTIS~~
~~A802 SYLOTI NAGRI SIGN DVISVARA~~
~~A806 SYLOTI NAGRI SIGN HASANTA~~
~~A80B SYLOTI NAGRI SIGN ANUSVARA~~
~~A823 SYLOTI NAGRI VOWEL SIGN A~~
~~A824 SYLOTI NAGRI VOWEL SIGN I~~
~~A825 SYLOTI NAGRI VOWEL SIGN U~~
~~A826 SYLOTI NAGRI VOWEL SIGN E~~
~~A827 SYLOTI NAGRI VOWEL SIGN OO~~
~~A880 SAURASHTRA SIGN ANUSVARA~~
~~A881 SAURASHTRA SIGN VISARGA~~
~~A8B4 SAURASHTRA CONSONANT SIGN HAARU~~
~~A8B5 SAURASHTRA VOWEL SIGN AA~~
~~A8B6 SAURASHTRA VOWEL SIGN I~~
~~A8B7 SAURASHTRA VOWEL SIGN II~~
~~A8B8 SAURASHTRA VOWEL SIGN U~~
~~A8B9 SAURASHTRA VOWEL SIGN UU~~
~~A8BA SAURASHTRA VOWEL SIGN VOCALIC R~~
~~A8BB SAURASHTRA VOWEL SIGN VOCALIC RR~~
~~A8BC SAURASHTRA VOWEL SIGN VOCALIC L~~
~~A8BD SAURASHTRA VOWEL SIGN VOCALIC LL~~
~~A8BE SAURASHTRA VOWEL SIGN E~~
~~A8BF SAURASHTRA VOWEL SIGN EE~~
~~A8C0 SAURASHTRA VOWEL SIGN AI~~

~~A8C1 SAURASHTRA VOWEL SIGN O~~
~~A8C2 SAURASHTRA VOWEL SIGN OO~~
~~A8C3 SAURASHTRA VOWEL SIGN AU~~
~~A8C4 SAURASHTRA SIGN VIRAMA~~
~~A926 KAYAH LI VOWEL UE~~
~~A927 KAYAH LI VOWEL E~~
~~A928 KAYAH LI VOWEL U~~
~~A929 KAYAH LI VOWEL EE~~
~~A92A KAYAH LI VOWEL O~~
~~A92B KAYAH LI TONE PLOPHU~~
~~A92C KAYAH LI TONE CALYA~~
~~A92D KAYAH LI TONE CALYA PLOPHU~~
~~A947 REJANG VOWEL SIGN I~~
~~A948 REJANG VOWEL SIGN U~~
~~A949 REJANG VOWEL SIGN E~~
~~A94A REJANG VOWEL SIGN AI~~
~~A94B REJANG VOWEL SIGN O~~
~~A94C REJANG VOWEL SIGN AU~~
~~A94D REJANG VOWEL SIGN EU~~
~~A94E REJANG VOWEL SIGN EA~~
~~A94F REJANG CONSONANT SIGN NG~~
~~A950 REJANG CONSONANT SIGN N~~
~~A951 REJANG CONSONANT SIGN R~~
~~A952 REJANG CONSONANT SIGN H~~
~~A953 REJANG VIRAMA~~
~~AA29 CHAM VOWEL SIGN AA~~
~~AA2A CHAM VOWEL SIGN I~~
~~AA2B CHAM VOWEL SIGN II~~
~~AA2C CHAM VOWEL SIGN EI~~
~~AA2D CHAM VOWEL SIGN U~~
~~AA2E CHAM VOWEL SIGN OE~~
~~AA2F CHAM VOWEL SIGN O~~
~~AA30 CHAM VOWEL SIGN AI~~
~~AA31 CHAM VOWEL SIGN AU~~
~~AA32 CHAM VOWEL SIGN UE~~
~~AA33 CHAM CONSONANT SIGN YA~~
~~AA34 CHAM CONSONANT SIGN RA~~
~~AA35 CHAM CONSONANT SIGN LA~~
~~AA36 CHAM CONSONANT SIGN WA~~
~~AA43 CHAM CONSONANT SIGN FINAL NG~~
~~AA4C CHAM CONSONANT SIGN FINAL M~~
~~AA4D CHAM CONSONANT SIGN FINAL H~~
~~AAB0 TAI VIET MAI KANG~~
~~AAB2 TAI VIET VOWEL I~~
~~AAB3 TAI VIET VOWEL UE~~
~~AAB4 TAI VIET VOWEL U~~
~~AAB7 TAI VIET MAY KHIT~~
~~AAB8 TAI VIET VOWEL IA~~
~~AABE TAI VIET VOWEL AM~~
~~AABF TAI VIET TONE MAI EK~~
~~AAC1 TAI VIET TONE MAI THO~~
~~FB1E HEBREW POINT JUDEO-SPANISH VARIKA~~
~~101FD PHAISTOS DISC SIGN COMBINING OBLIQUE STROKE~~
~~10A01 KHAROSHTHI VOWEL SIGN I~~
~~10A02 KHAROSHTHI VOWEL SIGN U~~
~~10A03 KHAROSHTHI VOWEL SIGN VOCALIC R~~
~~10A05 KHAROSHTHI VOWEL SIGN E~~
~~10A06 KHAROSHTHI VOWEL SIGN O~~
~~10A0C KHAROSHTHI VOWEL LENGTH MARK~~
~~10A0D KHAROSHTHI SIGN DOUBLE RING BELOW~~
~~10A0E KHAROSHTHI SIGN ANUSVARA~~
~~10A0F KHAROSHTHI SIGN VISARGA~~
~~10A38 KHAROSHTHI SIGN BAR ABOVE~~
~~10A39 KHAROSHTHI SIGN CAUDA~~
~~10A3A KHAROSHTHI SIGN DOT BELOW~~
~~1D165 MUSICAL SYMBOL COMBINING STEM~~

~~1D166 MUSICAL SYMBOL COMBINING SPRECHGESANG STEM~~
~~1D167 MUSICAL SYMBOL COMBINING TREMOLO ONE~~
~~1D168 MUSICAL SYMBOL COMBINING TREMOLO TWO~~
~~1D169 MUSICAL SYMBOL COMBINING TREMOLO THREE~~
~~1D16D MUSICAL SYMBOL COMBINING AUGMENTATION DOT~~
~~1D16E MUSICAL SYMBOL COMBINING FLAG ONE~~
~~1D16F MUSICAL SYMBOL COMBINING FLAG TWO~~
~~1D170 MUSICAL SYMBOL COMBINING FLAG THREE~~
~~1D171 MUSICAL SYMBOL COMBINING FLAG FOUR~~
~~1D172 MUSICAL SYMBOL COMBINING FLAG FIVE~~
~~1D17B MUSICAL SYMBOL COMBINING ACCENT~~
~~1D17C MUSICAL SYMBOL COMBINING STACCATO~~
~~1D17D MUSICAL SYMBOL COMBINING TENUTO~~
~~1D17E MUSICAL SYMBOL COMBINING STACCATISSIMO~~
~~1D17F MUSICAL SYMBOL COMBINING MARCATO~~
~~1D180 MUSICAL SYMBOL COMBINING MARCATO-STACCATO~~
~~1D181 MUSICAL SYMBOL COMBINING ACCENT-STACCATO~~
~~1D182 MUSICAL SYMBOL COMBINING LOURE~~
~~1D185 MUSICAL SYMBOL COMBINING DOIT~~
~~1D186 MUSICAL SYMBOL COMBINING RIP~~
~~1D187 MUSICAL SYMBOL COMBINING FLIP~~
~~1D188 MUSICAL SYMBOL COMBINING SMEAR~~
~~1D189 MUSICAL SYMBOL COMBINING BEND~~
~~1D18A MUSICAL SYMBOL COMBINING DOUBLE TONGUE~~
~~1D18B MUSICAL SYMBOL COMBINING TRIPLE TONGUE~~
~~1D1AA MUSICAL SYMBOL COMBINING DOWN BOW~~
~~1D1AB MUSICAL SYMBOL COMBINING UP BOW~~
~~1D1AC MUSICAL SYMBOL COMBINING HARMONIC~~
~~1D1AD MUSICAL SYMBOL COMBINING SNAP PIZZICATO~~
~~1D242 COMBINING GREEK MUSICAL TRISEME~~
~~1D243 COMBINING GREEK MUSICAL TETRASEME~~
~~1D244 COMBINING GREEK MUSICAL PENTASEME~~

Annex B
(normative)
List of combining characters

NOTE – Replaced by formal character class definition, see 4.14

Annex C
(normative)
Transformation format for planes 1 to 10 of the UCS (UTF-16)

NOTE – Incorporated in main body text, see UCS UTF-16 encoding form in 9 and UCS UTF-16 based encoding schemes in 10.

~~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 (see 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 (see 4.38), 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.~~

~~When used for serialization purpose, UTF-16 does not specify the ordering of the octets; a signature may be used (see Annex H).~~

~~Two additional UCS Transformation Formats, derived from UTF-16, are specified for serialization purpose.~~

~~1) 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;~~

~~2) 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) 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 7). 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, y = DC00...DFFF ((x - D800) * 400 + (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 𐄀!!" (𐄀 is the graphic symbol representing 10000 LINEAR B SYLLABLE B008 A).~~

~~C.5 Identification of UTF-16~~

~~When the escape sequences from ISO/IEC 2022 are used, the identification of UTF-16 shall be by the following designation sequence:~~

~~ESC 02/05 02/15 04/12 UTF-16~~

~~NOTE – The following designation sequences: ESC 02/05 02/15 04/10 and ESC 02/05 02/15 04/11 used in previous versions of this standard to identify implementation levels 1 and 2 are deprecated. The remaining designation sequence corresponds to the former level 3 which is now the only supported CC-data-element content definition.~~

~~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 clause 16.5 for a return or transfer from UCS.~~

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

~~According to clause C.1 an unpaired RC-element (see 4.46) 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 (see 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.~~

~~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. Consequently 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:~~

~~a) 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?~~

~~b) 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 ◊◊ and to ◊◊."~~

~~W: "The Greek letter Σ corresponds to ◊◊ and to Σ."~~

~~A: "The Greek letter Σ corresponds to ◊ and to ◊."~~

~~S: "The Greek letter Σ corresponds to ◊ and to Σ."~~

~~where Σ 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 to 001F, and the DELETE character in position 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
1st of 1	0xxxxxxx	7	0000 007F
1st of 2	110xxxxx	5	0000 07FF
1st of 3	1110xxxx	4	0000 FFFF
1st of 4	11110xxx	3	001F FFFF
1st of 5	111110xx	2	03FF FFFF
1st of 6	1111110x	1	7FFF FFFF
continuing) 2nd ... 6th)	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.

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; 10000000;
01111111 11111111 11111111 11111111;	11111101; 10111111; 10111111;10111111; 10111111; 10111111;

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 - 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.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.

~~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 code position of the UCS. In the right column (UTF-8) x indicates the corresponding integer value.~~

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

~~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;~~	~~$FC + x/2^{30}$;~~ ~~$80 + x/2^{24} \% 2^6$;~~ ~~$80 + x/2^{18} \% 2^6$;~~ ~~$80 + x/2^{12} \% 2^6$;~~ ~~$80 + x/2^6 \% 2^6$;~~ ~~$80 + x \% 2^6$;~~

~~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 code position 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.~~

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

~~Sequence of octets in UTF-8~~	~~Four-octet sequences in UCS-4~~
~~z = 00 .. 7F;~~	~~z;~~
~~z = C0 .. DF; y;~~	~~(z-C0)*2^6 + (y-80);~~
~~z = E0 .. EF; y; x;~~	~~(z-E0)*2^12 + (y-80)*2^6 + (x-80);~~
~~z = F0 .. F7; y; x; w;~~	~~(z-F0)*2^18 + (y-80)*2^12 +(x-80)*2^6 + (w-80);~~
~~z = F8 ..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);~~

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

~~For each octet 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.~~

~~D.6 Identification of UTF-8~~

~~When the escape sequences from ISO/IEC 2022 are used, the identification of UTF-8 shall be by the following designation sequence:~~

~~ESC 02/05 02/15 04/09~~
~~UTF-8~~

~~NOTE – The following designation sequences: ESC 02/05 02/15 04/07 and ESC 02/05 02/15 04/08 used in previous versions of this standard to identify implementation levels 1 and 2 are deprecated. The remaining designation sequence corresponds to the former level 3 which is now the only supported CC-data-element content definition.~~

~~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 escape sequence used for a return to the coding system of ISO/IEC 2022 is ESC 02/05 04/00 and is not padded (seen 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).~~NOTE – Incorporated in main body text, see UCS UTF-8 encoding form in 9 and UCS UTF-8 encoding schemes in 10.

Annex E
(normative)
Mirrored characters in bidirectional context

NOTE – Replaced by formal character class definition for mirrored character, see 15.1.

~~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.~~

~~NOTE – Many ancient scripts and some scripts in modern use can be written either right-to-left or left-to-right. It is often customary for one of these scripts to use the appropriately mirrored graphical symbol for any character represented by a graphic symbol that is not symmetric around the vertical axis. In such cases, it is up to the rendering system to display the graphic image appropriate for the writing direction employed. The directionality of the representative graphic symbol shown in the character code charts matches the default writing direction for the script.~~

~~Examples of such scripts include, but are not limited to, Old Italic, an ancient script for which the default writing direction in this standard is left-to-right, and Cypriot, an ancient script for which the default writing direction in this standard is right-to-left.~~

~~0028 LEFT PARENTHESIS~~
~~0029 RIGHT PARENTHESIS~~
~~003C LESS-THAN SIGN~~
~~003E GREATER-THAN SIGN~~
~~005B LEFT SQUARE BRACKET~~
~~005D RIGHT SQUARE BRACKET~~
~~007B LEFT CURLY BRACKET~~
~~007D RIGHT CURLY BRACKET~~
~~00AB LEFT-POINTING DOUBLE ANGLE QUOTATION MARK~~
~~00BB RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK~~
~~0F3A TIBETAN MARK GUG RTAGS GYON~~
~~0F3B TIBETAN MARK GUG RTAGS GYAS~~
~~0F3C TIBETAN MARK ANG KHANG GYON~~
~~0F3D TIBETAN MARK ANG KHANG GYAS~~
~~169B OGHAM FEATHER MARK~~
~~169C OGHAM REVERSED FEATHER MARK~~
~~2018 LEFT SINGLE QUOTATION MARK~~
~~2019 RIGHT SINGLE QUOTATION MARK~~
~~201A SINGLE LOW-9 QUOTATION MARK~~
~~201B SINGLE HIGH-REVERSED-9 QUOTATION MARK~~
~~201C LEFT DOUBLE QUOTATION MARK~~
~~201D RIGHT DOUBLE QUOTATION MARK~~
~~201E DOUBLE LOW-9 QUOTATION MARK~~
~~201F DOUBLE HIGH-REVERSED-9 QUOTATION MARK~~
~~2039 SINGLE LEFT-POINTING ANGLE QUOTATION MARK~~
~~203A SINGLE RIGHT-POINTING ANGLE QUOTATION MARK~~
~~2045 LEFT SQUARE BRACKET WITH QUILL~~
~~2046 RIGHT SQUARE BRACKET WITH QUILL~~
~~207D SUPERSCRIPT LEFT PARENTHESIS~~
~~207E SUPERSCRIPT RIGHT PARENTHESIS~~
~~208D SUBSCRIPT LEFT PARENTHESIS~~
~~208E SUBSCRIPT RIGHT PARENTHESIS~~
~~2140 DOUBLE-STRUCK N-ARY SUMMATION~~
~~2201 COMPLEMENT~~
~~2202 PARTIAL DIFFERENTIAL~~
~~2203 THERE EXISTS~~
~~2204 THERE DOES NOT EXIST~~
~~2208 ELEMENT OF~~
~~2209 NOT AN ELEMENT OF~~
~~220A SMALL ELEMENT OF~~
~~220B CONTAINS AS MEMBER~~
~~220C DOES NOT CONTAIN AS MEMBER~~
~~220D SMALL CONTAINS AS MEMBER~~
~~2211 N-ARY SUMMATION~~
~~2215 DIVISION SLASH~~
~~2216 SET MINUS~~
~~221A SQUARE ROOT~~
~~221B CUBE ROOT~~
~~221C FOURTH ROOT~~
~~221D PROPORTIONAL TO~~
~~221F RIGHT ANGLE~~
~~2220 ANGLE~~
~~2221 MEASURED ANGLE~~
~~2222 SPHERICAL ANGLE~~
~~2224 DOES NOT DIVIDE~~
~~2226 NOT PARALLEL TO~~
~~222B INTEGRAL~~
~~222C DOUBLE INTEGRAL~~
~~222D TRIPLE INTEGRAL~~
~~222E CONTOUR INTEGRAL~~
~~222F SURFACE INTEGRAL~~
~~2230 VOLUME INTEGRAL~~
~~2231 CLOCKWISE INTEGRAL~~
~~2232 CLOCKWISE CONTOUR INTEGRAL~~
~~2233 ANTICLOCKWISE CONTOUR INTEGRAL~~
~~2239 EXCESS~~
~~223B HOMOTHETIC~~
~~223C TILDE OPERATOR~~
~~223D REVERSED TILDE~~
~~223E INVERTED LAZY S~~
~~223F SINE WAVE~~
~~2240 WREATH PRODUCT~~
~~2241 NOT TILDE~~
~~2242 MINUS TILDE~~
~~2243 ASYMPTOTICALLY EQUAL TO~~
~~2244 NOT ASYMPTOTICALLY EQUAL TO~~
~~2245 APPROXIMATELY EQUAL TO~~
~~2246 APPROXIMATELY BUT NOT ACTUALLY EQUAL TO~~
~~2247 NEITHER APPROXIMATELY NOR ACTUALLY EQUAL TO~~
~~2248 ALMOST EQUAL TO~~
~~2249 NOT ALMOST EQUAL TO~~
~~224A ALMOST EQUAL OR EQUAL TO~~
~~224B TRIPLE TILDE~~

~~224C ALL EQUAL TO~~
~~2252 APPROXIMATELY EQUAL TO OR THE IMAGE OF~~
~~2253 IMAGE OF OR APPROXIMATELY EQUAL TO~~
~~2254 COLON EQUALS~~
~~2255 EQUALS COLON~~
~~225F QUESTIONED EQUAL TO~~
~~2260 NOT EQUAL TO~~
~~2262 NOT IDENTICAL TO~~
~~2264 LESS-THAN OR EQUAL TO~~
~~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~~
~~2275 NEITHER GREATER-THAN NOR EQUIVALENT TO~~
~~2276 LESS-THAN OR GREATER-THAN~~
~~2277 GREATER-THAN OR LESS-THAN~~
~~2278 NEITHER LESS-THAN NOR GREATER-THAN~~
~~2279 NEITHER GREATER-THAN NOR LESS-THAN~~
~~227A PRECEDES~~
~~227B SUCCEEDS~~
~~227C PRECEDES OR EQUAL TO~~
~~227D SUCCEEDS OR EQUAL TO~~
~~227E PRECEDES OR EQUIVALENT TO~~
~~227F SUCCEEDS OR EQUIVALENT TO~~
~~2280 DOES NOT PRECEDE~~
~~2281 DOES NOT SUCCEED~~
~~2282 SUBSET OF~~
~~2283 SUPERSET OF~~
~~2284 NOT A SUBSET OF~~
~~2285 NOT A SUPERSET OF~~
~~2286 SUBSET OF OR EQUAL TO~~
~~2287 SUPERSET OF OR EQUAL TO~~
~~2288 NEITHER A SUBSET OF NOR EQUAL TO~~
~~2289 NEITHER A SUPERSET OF NOR EQUAL TO~~
~~228A SUBSET OF WITH NOT EQUAL TO~~
~~228B SUPERSET OF WITH NOT EQUAL TO~~
~~228C MULTISET~~
~~228F SQUARE IMAGE OF~~
~~2290 SQUARE ORIGINAL OF~~
~~2291 SQUARE IMAGE OF OR EQUAL TO~~
~~2292 SQUARE ORIGINAL OF OR EQUAL TO~~
~~2298 CIRCLED DIVISION SLASH~~
~~22A2 RIGHT TACK~~
~~22A3 LEFT TACK~~
~~22A6 ASSERTION~~
~~22A7 MODELS~~
~~22A8 TRUE~~
~~22A9 FORCES~~
~~22AA TRIPLE VERTICAL BAR RIGHT TURNSTILE~~
~~22AB DOUBLE VERTICAL BAR DOUBLE RIGHT TURNSTILE~~
~~22AC DOES NOT PROVE~~
~~22AD NOT TRUE~~
~~22AE DOES NOT FORCE~~
~~22AF NEGATED DOUBLE VERTICAL BAR DOUBLE RIGHT TURNSTILE~~
~~22B0 PRECEDES UNDER RELATION~~
~~22B1 SUCCEEDS UNDER RELATION~~
~~22B2 NORMAL SUBGROUP OF~~
~~22B3 CONTAINS AS NORMAL SUBGROUP~~
~~22B4 NORMAL SUBGROUP OF OR EQUAL TO~~
~~22B5 CONTAINS AS NORMAL SUBGROUP OR EQUAL TO~~
~~22B6 ORIGINAL OF~~
~~22B7 IMAGE OF~~
~~22B8 MULTIMAP~~
~~22BE RIGHT ANGLE WITH ARC~~
~~22BF RIGHT TRIANGLE~~
~~22C9 LEFT NORMAL FACTOR SEMIDIRECT PRODUCT~~
~~22CA RIGHT NORMAL FACTOR SEMIDIRECT PRODUCT~~
~~22CB LEFT SEMIDIRECT PRODUCT~~
~~22CC RIGHT SEMIDIRECT PRODUCT~~
~~22CD REVERSED TILDE EQUALS~~
~~22D0 DOUBLE SUBSET~~
~~22D1 DOUBLE SUPERSET~~
~~22D6 LESS-THAN WITH DOT~~
~~22D7 GREATER-THAN WITH DOT~~
~~22D8 VERY MUCH LESS-THAN~~
~~22D9 VERY MUCH GREATER-THAN~~
~~22DA LESS-THAN EQUAL TO OR GREATER-THAN~~
~~22DB GREATER-THAN EQUAL TO OR LESS-THAN~~
~~22DC EQUAL TO OR LESS-THAN~~
~~22DD EQUAL TO OR GREATER-THAN~~
~~22DE EQUAL TO OR PRECEDES~~
~~22DF EQUAL TO OR SUCCEEDS~~
~~22E0 DOES NOT PRECEDE OR EQUAL~~
~~22E1 DOES NOT SUCCEED OR EQUAL~~
~~22E2 NOT SQUARE IMAGE OF OR EQUAL TO~~
~~22E3 NOT SQUARE ORIGINAL OF OR EQUAL TO~~
~~22E4 SQUARE IMAGE OF OR NOT EQUAL TO~~
~~22E5 SQUARE ORIGINAL OF OR NOT EQUAL TO~~
~~22E6 LESS-THAN BUT NOT EQUIVALENT TO~~
~~22E7 GREATER-THAN BUT NOT EQUIVALENT TO~~
~~22E8 PRECEDES BUT NOT EQUIVALENT TO~~
~~22E9 SUCCEEDS BUT NOT EQUIVALENT TO~~
~~22EA NOT NORMAL SUBGROUP OF~~
~~22EB DOES NOT CONTAIN AS NORMAL SUBGROUP~~
~~22EC NOT NORMAL SUBGROUP OF OR EQUAL TO~~
~~22ED DOES NOT CONTAIN AS NORMAL SUBGROUP OR EQUAL~~
~~22F0 UP RIGHT DIAGONAL ELLIPSIS~~
~~22F1 DOWN RIGHT DIAGONAL ELLIPSIS~~
~~22F2 ELEMENT OF WITH LONG HORIZONTAL STROKE~~
~~22F3 ELEMENT OF WITH VERTICAL BAR AT END OF HORIZONTAL STROKE~~
~~22F4 SMALL ELEMENT OF WITH VERTICAL BAR AT END OF HORIZONTAL STROKE~~
~~22F5 ELEMENT OF WITH DOT ABOVE~~
~~22F6 ELEMENT OF WITH OVERBAR~~
~~22F7 SMALL ELEMENT OF WITH OVERBAR~~
~~22F8 ELEMENT OF WITH UNDERBAR~~
~~22F9 ELEMENT OF WITH TWO HORIZONTAL STROKES~~
~~22FA CONTAINS WITH LONG HORIZONTAL STROKE~~
~~22FB CONTAINS WITH VERTICAL BAR AT END OF HORIZONTAL STROKE~~
~~22FC SMALL CONTAINS WITH VERTICAL BAR AT END OF HORIZONTAL STROKE~~
~~22FD CONTAINS WITH OVERBAR~~
~~22FE SMALL CONTAINS WITH OVERBAR~~
~~22FF Z NOTATION BAG MEMBERSHIP~~
~~2308 LEFT CEILING~~

~~2309 RIGHT CEILING~~
~~230A LEFT FLOOR~~
~~230B RIGHT FLOOR~~
~~2320 TOP HALF INTEGRAL~~
~~2321 BOTTOM HALF INTEGRAL~~
~~2329 LEFT-POINTING ANGLE BRACKET~~
~~232A RIGHT-POINTING ANGLE BRACKET~~
~~2768 MEDIUM LEFT PARENTHESIS ORNAMENT~~
~~2769 MEDIUM RIGHT PARENTHESIS ORNAMENT~~
~~276A MEDIUM FLATTENED LEFT PARENTHESIS ORNAMENT~~
~~276B MEDIUM FLATTENED RIGHT PARENTHESIS ORNAMENT~~
~~276C MEDIUM LEFT-POINTING ANGLE BRACKET ORNAMENT~~
~~276D MEDIUM RIGHT-POINTING ANGLE BRACKET ORNAMENT~~
~~276E HEAVY LEFT-POINTING ANGLE QUOTATION MARK ORNAMENT~~
~~276F HEAVY RIGHT-POINTING ANGLE QUOTATION MARK ORNAMENT~~
~~2770 HEAVY LEFT-POINTING ANGLE BRACKET ORNAMENT~~
~~2771 HEAVY RIGHT-POINTING ANGLE BRACKET ORNAMENT~~
~~2772 LIGHT LEFT TORTOISE SHELL BRACKET ORNAMENT~~
~~2773 LIGHT RIGHT TORTOISE SHELL BRACKET ORNAMENT~~
~~2774 MEDIUM LEFT CURLY BRACKET ORNAMENT~~
~~2775 MEDIUM RIGHT CURLY BRACKET ORNAMENT~~
~~27C0 THREE DIMENSIONAL ANGLE~~
~~27C3 OPEN SUBSET~~
~~27C4 OPEN SUPERSET~~
~~27C5 LEFT S-SHAPED BAG DELIMITER~~
~~27C6 RIGHT S-SHAPED BAG DELIMITER~~
~~27D3 LOWER RIGHT CORNER WITH DOT~~
~~27D4 UPPER LEFT CORNER WITH DOT~~
~~27D5 LEFT OUTER JOIN~~
~~27D6 RIGHT OUTER JOIN~~
~~27DC LEFT MULTIMAP~~
~~27DD LONG RIGHT TACK~~
~~27DE LONG LEFT TACK~~
~~27E2 WHITE CONCAVE-SIDED DIAMOND WITH LEFTWARDS TICK~~
~~27E3 WHITE CONCAVE-SIDED DIAMOND WITH RIGHTWARDS TICK~~
~~27E4 WHITE SQUARE WITH LEFTWARDS TICK~~
~~27E5 WHITE SQUARE WITH RIGHTWARDS TICK~~
~~27E6 MATHEMATICAL LEFT WHITE SQUARE BRACKET~~
~~27E7 MATHEMATICAL RIGHT WHITE SQUARE BRACKET~~
~~27E8 MATHEMATICAL LEFT ANGLE BRACKET~~
~~27E9 MATHEMATICAL RIGHT ANGLE BRACKET~~
~~27EA MATHEMATICAL LEFT DOUBLE ANGLE BRACKET~~
~~27EB MATHEMATICAL RIGHT DOUBLE ANGLE BRACKET~~
~~27EC MATHEMATICAL LEFT WHITE TORTOISE SHELL BRACKET~~
~~27ED MATHEMATICAL RIGHT WHITE TORTOISE SHELL BRACKET~~
~~2983 LEFT WHITE CURLY BRACKET~~
~~2984 RIGHT WHITE CURLY BRACKET~~
~~2985 LEFT WHITE PARENTHESIS~~
~~2986 RIGHT WHITE PARENTHESIS~~
~~2987 Z NOTATION LEFT IMAGE BRACKET~~
~~2988 Z NOTATION RIGHT IMAGE BRACKET~~
~~2989 Z NOTATION LEFT BINDING BRACKET~~
~~298A Z NOTATION RIGHT BINDING BRACKET~~
~~298B LEFT SQUARE BRACKET WITH UNDERBAR~~
~~298C RIGHT SQUARE BRACKET WITH UNDERBAR~~
~~298D LEFT SQUARE BRACKET WITH TICK IN TOP CORNER~~
~~298E RIGHT SQUARE BRACKET WITH TICK IN BOTTOM CORNER~~
~~298F LEFT SQUARE BRACKET WITH TICK IN BOTTOM CORNER~~
~~2990 RIGHT SQUARE BRACKET WITH TICK IN TOP CORNER~~
~~2991 LEFT ANGLE BRACKET WITH DOT~~
~~2992 RIGHT ANGLE BRACKET WITH DOT~~
~~2993 LEFT ARC LESS-THAN BRACKET~~
~~2994 RIGHT ARC GREATER-THAN BRACKET~~
~~2995 DOUBLE LEFT ARC GREATER-THAN BRACKET~~
~~2996 DOUBLE RIGHT ARC LESS-THAN BRACKET~~
~~2997 LEFT BLACK TORTOISE SHELL BRACKET~~
~~2998 RIGHT BLACK TORTOISE SHELL BRACKET~~
~~299B MEASURED ANGLE OPENING LEFT~~
~~299C RIGHT ANGLE VARIANT WITH SQUARE~~
~~299D MEASURED RIGHT ANGLE WITH DOT~~
~~299E ANGLE WITH S INSIDE~~
~~299F ACUTE ANGLE~~
~~29A0 SPHERICAL ANGLE OPENING LEFT~~
~~29A1 SPHERICAL ANGLE OPENING UP~~
~~29A2 TURNED ANGLE~~
~~29A3 REVERSED ANGLE~~
~~29A4 ANGLE WITH UNDERBAR~~
~~29A5 REVERSED ANGLE WITH UNDERBAR~~
~~29A6 OBLIQUE ANGLE OPENING UP~~
~~29A7 OBLIQUE ANGLE OPENING DOWN~~
~~29A8 MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING UP AND RIGHT~~
~~29A9 MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING UP AND LEFT~~
~~29AA MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING DOWN AND RIGHT~~
~~29AB MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING DOWN AND LEFT~~
~~29AC MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING RIGHT AND UP~~
~~29AD MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING LEFT AND UP~~
~~29AE MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING RIGHT AND DOWN~~
~~29AF MEASURED ANGLE WITH OPEN ARM ENDING IN ARROW POINTING LEFT AND DOWN~~
~~29B8 CIRCLED REVERSE SOLIDUS~~
~~29C0 CIRCLED LESS-THAN~~
~~29C1 CIRCLED GREATER-THAN~~
~~29C2 CIRCLE WITH SMALL CIRCLE TO THE RIGHT~~
~~29C3 CIRCLE WITH TWO HORIZONTAL STROKES TO THE RIGHT~~
~~29C4 SQUARED RISING DIAGONAL SLASH~~
~~29C5 SQUARED FALLING DIAGONAL SLASH~~
~~29C9 TWO JOINED SQUARES~~
~~29CE RIGHT TRIANGLE ABOVE LEFT TRIANGLE~~
~~29CF LEFT TRIANGLE BESIDE VERTICAL BAR~~
~~29D0 VERTICAL BAR BESIDE RIGHT TRIANGLE~~
~~29D1 BOWTIE WITH LEFT HALF BLACK~~
~~29D2 BOWTIE WITH RIGHT HALF BLACK~~
~~29D4 TIMES WITH LEFT HALF BLACK~~
~~29D5 TIMES WITH RIGHT HALF BLACK~~

~~29D8 LEFT WIGGLY FENCE~~
~~29D9 RIGHT WIGGLY FENCE~~
~~29DA LEFT DOUBLE WIGGLY FENCE~~
~~29DB RIGHT DOUBLE WIGGLY FENCE~~
~~29DC INCOMPLETE INFINITY~~
~~29E1 INCREASES AS~~
~~29E3 EQUALS SIGN AND SLANTED PARALLEL~~
~~29E4 EQUALS SIGN AND SLANTED PARALLEL WITH TILDE ABOVE~~
~~29E5 IDENTICAL TO AND SLANTED PARALLEL~~
~~29E8 DOWN-POINTING TRIANGLE WITH LEFT HALF BLACK~~
~~29E9 DOWN-POINTING TRIANGLE WITH RIGHT HALF BLACK~~
~~29F4 RULE-DELAYED~~
~~29F5 REVERSE SOLIDUS OPERATOR~~
~~29F6 SOLIDUS WITH OVERBAR~~
~~29F7 REVERSE SOLIDUS WITH HORIZONTAL STROKE~~
~~29F8 BIG SOLIDUS~~
~~29F9 BIG REVERSE SOLIDUS~~
~~29FC LEFT-POINTING CURVED ANGLE BRACKET~~
~~29FD RIGHT-POINTING CURVED ANGLE BRACKET~~
~~2A0A MODULO TWO SUM~~
~~2A0B SUMMATION WITH INTEGRAL~~
~~2A0C QUADRUPLE INTEGRAL OPERATOR~~
~~2A0D FINITE PART INTEGRAL~~
~~2A0E INTEGRAL WITH DOUBLE STROKE~~
~~2A0F INTEGRAL AVERAGE WITH SLASH~~
~~2A10 CIRCULATION FUNCTION~~
~~2A11 ANTICLOCKWISE INTEGRATION~~
~~2A12 LINE INTEGRATION WITH RECTANGULAR PATH AROUND POLE~~
~~2A13 LINE INTEGRATION WITH SEMICIRCULAR PATH AROUND POLE~~
~~2A14 LINE INTEGRATION NOT INCLUDING THE POLE~~
~~2A15 INTEGRAL AROUND A POINT OPERATOR~~
~~2A16 QUATERNION INTEGRAL OPERATOR~~
~~2A17 INTEGRAL WITH LEFTWARDS ARROW WITH HOOK~~
~~2A18 INTEGRAL WITH TIMES SIGN~~
~~2A19 INTEGRAL WITH INTERSECTION~~
~~2A1A INTEGRAL WITH UNION~~
~~2A1B INTEGRAL WITH OVERBAR~~
~~2A1C INTEGRAL WITH UNDERBAR~~
~~2A1E LARGE LEFT TRIANGLE OPERATOR~~
~~2A1F Z NOTATION SCHEMA COMPOSITION~~
~~2A20 Z NOTATION SCHEMA PIPING~~
~~2A21 Z NOTATION SCHEMA PROJECTION~~
~~2A24 PLUS SIGN WITH TILDE ABOVE~~
~~2A26 PLUS SIGN WITH TILDE BELOW~~
~~2A29 MINUS SIGN WITH COMMA ABOVE~~
~~2A2B MINUS SIGN WITH FALLING DOTS~~
~~2A2C MINUS SIGN WITH RISING DOTS~~
~~2A2D PLUS SIGN IN LEFT HALF CIRCLE~~
~~2A2E PLUS SIGN IN RIGHT HALF CIRCLE~~
~~2A34 MULTIPLICATION SIGN IN LEFT HALF CIRCLE~~
~~2A35 MULTIPLICATION SIGN IN RIGHT HALF CIRCLE~~
~~2A3C INTERIOR PRODUCT~~
~~2A3D RIGHTHAND INTERIOR PRODUCT~~
~~2A3E Z NOTATION RELATIONAL COMPOSITION~~
~~2A57 SLOPING LARGE OR~~
~~2A58 SLOPING LARGE AND~~
~~2A64 Z NOTATION DOMAIN ANTIRESTRICTION~~
~~2A65 Z NOTATION RANGE ANTIRESTRICTION~~
~~2A6A TILDE OPERATOR WITH DOT ABOVE~~
~~2A6B TILDE OPERATOR WITH RISING DOTS~~
~~2A6C SIMILAR MINUS SIMILAR~~
~~2A6D CONGRUENT WITH DOT ABOVE~~
~~2A6F ALMOST EQUAL TO WITH CIRCUMFLEX ACCENT~~
~~2A70 APPROXIMATELY EQUAL OR EQUAL TO~~
~~2A73 EQUALS SIGN ABOVE TILDE OPERATOR~~
~~2A74 DOUBLE COLON EQUAL~~
~~2A79 LESS-THAN WITH CIRCLE INSIDE~~
~~2A7A GREATER-THAN WITH CIRCLE INSIDE~~
~~2A7B LESS-THAN WITH QUESTION MARK ABOVE~~
~~2A7C GREATER-THAN WITH QUESTION MARK ABOVE~~
~~2A7D LESS-THAN OR SLANTED EQUAL TO~~
~~2A7E GREATER-THAN OR SLANTED EQUAL TO~~
~~2A7F LESS-THAN OR SLANTED EQUAL TO WITH DOT INSIDE~~
~~2A80 GREATER-THAN OR SLANTED EQUAL TO WITH DOT INSIDE~~
~~2A81 LESS-THAN OR SLANTED EQUAL TO WITH DOT ABOVE~~
~~2A82 GREATER-THAN OR SLANTED EQUAL TO WITH DOT ABOVE~~
~~2A83 LESS-THAN OR SLANTED EQUAL TO WITH DOT ABOVE RIGHT~~
~~2A84 GREATER-THAN OR SLANTED EQUAL TO WITH DOT ABOVE LEFT~~
~~2A85 LESS-THAN OR APPROXIMATE~~
~~2A86 GREATER-THAN OR APPROXIMATE~~
~~2A87 LESS-THAN AND SINGLE-LINE NOT EQUAL TO~~
~~2A88 GREATER-THAN AND SINGLE-LINE NOT EQUAL TO~~
~~2A89 LESS-THAN AND NOT APPROXIMATE~~
~~2A8A GREATER-THAN AND NOT APPROXIMATE~~
~~2A8B LESS-THAN ABOVE DOUBLE-LINE EQUAL ABOVE GREATER-THAN~~
~~2A8C GREATER-THAN ABOVE DOUBLE-LINE EQUAL ABOVE LESS-THAN~~
~~2A8D LESS-THAN ABOVE SIMILAR OR EQUAL~~
~~2A8E GREATER-THAN ABOVE SIMILAR OR EQUAL~~
~~2A8F LESS-THAN ABOVE SIMILAR ABOVE GREATER-THAN~~
~~2A90 GREATER-THAN ABOVE SIMILAR ABOVE LESS-THAN~~
~~2A91 LESS-THAN ABOVE GREATER-THAN ABOVE DOUBLE-LINE EQUAL~~
~~2A92 GREATER-THAN ABOVE LESS-THAN ABOVE DOUBLE-LINE EQUAL~~
~~2A93 LESS-THAN ABOVE SLANTED EQUAL ABOVE GREATER-THAN ABOVE SLANTED EQUAL~~
~~2A94 GREATER-THAN ABOVE SLANTED EQUAL ABOVE LESS-THAN ABOVE SLANTED EQUAL~~
~~2A95 SLANTED EQUAL TO OR LESS-THAN~~
~~2A96 SLANTED EQUAL TO OR GREATER-THAN~~
~~2A97 SLANTED EQUAL TO OR LESS-THAN WITH DOT INSIDE~~
~~2A98 SLANTED EQUAL TO OR GREATER-THAN WITH DOT INSIDE~~
~~2A99 DOUBLE-LINE EQUAL TO OR LESS-THAN~~
~~2A9A DOUBLE-LINE EQUAL TO OR GREATER-THAN~~
~~2A9B DOUBLE-LINE SLANTED EQUAL TO OR LESS-THAN~~
~~2A9C DOUBLE-LINE SLANTED EQUAL TO OR GREATER-THAN~~
~~2A9D SIMILAR OR LESS-THAN~~
~~2A9E SIMILAR OR GREATER-THAN~~
~~2A9F SIMILAR ABOVE LESS-THAN ABOVE EQUALS SIGN~~

~~2AA0 SIMILAR ABOVE GREATER-THAN ABOVE EQUALS SIGN~~
~~2AA1 DOUBLE NESTED LESS-THAN~~
~~2AA2 DOUBLE NESTED GREATER-THAN~~
~~2AA3 DOUBLE NESTED LESS-THAN WITH UNDERBAR~~
~~2AA6 LESS-THAN CLOSED BY CURVE~~
~~2AA7 GREATER-THAN CLOSED BY CURVE~~
~~2AA8 LESS-THAN CLOSED BY CURVE ABOVE SLANTED EQUAL~~
~~2AA9 GREATER-THAN CLOSED BY CURVE ABOVE SLANTED EQUAL~~
~~2AAA SMALLER THAN~~
~~2AAB LARGER THAN~~
~~2AAC SMALLER THAN OR EQUAL TO~~
~~2AAD LARGER THAN OR EQUAL TO~~
~~2AAF PRECEDES ABOVE SINGLE-LINE EQUALS SIGN~~
~~2AB0 SUCCEEDS ABOVE SINGLE-LINE EQUALS SIGN~~
~~2AB1 PRECEDES ABOVE SINGLE-LINE NOT EQUAL TO~~
~~2AB2 SUCCEEDS ABOVE SINGLE-LINE NOT EQUAL TO~~
~~2AB3 PRECEDES ABOVE EQUALS SIGN~~
~~2AB4 SUCCEEDS ABOVE EQUALS SIGN~~
~~2AB5 PRECEDES ABOVE NOT EQUAL TO~~
~~2AB6 SUCCEEDS ABOVE NOT EQUAL TO~~
~~2AB7 PRECEDES ABOVE ALMOST EQUAL TO~~
~~2AB8 SUCCEEDS ABOVE ALMOST EQUAL TO~~
~~2AB9 PRECEDES ABOVE NOT ALMOST EQUAL TO~~
~~2ABA SUCCEEDS ABOVE NOT ALMOST EQUAL TO~~
~~2ABB DOUBLE PRECEDES~~
~~2ABC DOUBLE SUCCEEDS~~
~~2ABD SUBSET WITH DOT~~
~~2ABE SUPERSET WITH DOT~~
~~2ABF SUBSET WITH PLUS SIGN BELOW~~
~~2AC0 SUPERSET WITH PLUS SIGN BELOW~~
~~2AC1 SUBSET WITH MULTIPLICATION SIGN BELOW~~
~~2AC2 SUPERSET WITH MULTIPLICATION SIGN BELOW~~
~~2AC3 SUBSET OF OR EQUAL TO WITH DOT ABOVE~~
~~2AC4 SUPERSET OF OR EQUAL TO WITH DOT ABOVE~~
~~2AC5 SUBSET OF ABOVE EQUALS SIGN~~
~~2AC6 SUPERSET OF ABOVE EQUALS SIGN~~
~~2AC7 SUBSET OF ABOVE TILDE OPERATOR~~
~~2AC8 SUPERSET OF ABOVE TILDE OPERATOR~~
~~2AC9 SUBSET OF ABOVE ALMOST EQUAL TO~~
~~2ACA SUPERSET OF ABOVE ALMOST EQUAL TO~~
~~2ACB SUBSET OF ABOVE NOT EQUAL TO~~
~~2ACC SUPERSET OF ABOVE NOT EQUAL TO~~
~~2ACD SQUARE LEFT OPEN BOX OPERATOR~~
~~2ACE SQUARE RIGHT OPEN BOX OPERATOR~~
~~2ACF CLOSED SUBSET~~
~~2AD0 CLOSED SUPERSET~~
~~2AD1 CLOSED SUBSET OR EQUAL TO~~
~~2AD2 CLOSED SUPERSET OR EQUAL TO~~
~~2AD3 SUBSET ABOVE SUPERSET~~
~~2AD4 SUPERSET ABOVE SUBSET~~
~~2AD5 SUBSET ABOVE SUBSET~~
~~2AD6 SUPERSET ABOVE SUPERSET~~
~~2ADC FORKING~~
~~2ADE SHORT LEFT TACK~~
~~2AE2 VERTICAL BAR TRIPLE RIGHT TURNSTILE~~
~~2AE3 DOUBLE VERTICAL BAR LEFT TURNSTILE~~
~~2AE4 VERTICAL BAR DOUBLE LEFT TURNSTILE~~
~~2AE5 DOUBLE VERTICAL BAR DOUBLE LEFT TURNSTILE~~
~~2AE6 LONG DASH FROM LEFT MEMBER OF DOUBLE VERTICAL~~
~~2AEC DOUBLE STROKE NOT SIGN~~
~~2AED REVERSED DOUBLE STROKE NOT SIGN~~
~~2AEE DOES NOT DIVIDE WITH REVERSED NEGATION SLASH~~
~~2AF3 PARALLEL WITH TILDE OPERATOR~~
~~2AF7 TRIPLE NESTED LESS-THAN~~
~~2AF8 TRIPLE NESTED GREATER-THAN~~
~~2AF9 DOUBLE-LINE SLANTED LESS-THAN OR EQUAL TO~~
~~2AFA DOUBLE-LINE SLANTED GREATER-THAN OR EQUAL TO~~
~~2AFB TRIPLE SOLIDUS BINARY RELATION~~
~~2AFD DOUBLE SOLIDUS OPERATOR~~
~~2E02 LEFT SUBSTITUTION BRACKET~~
~~2E03 RIGHT SUBSTITUTION BRACKET~~
~~2E04 LEFT DOTTED SUBSTITUTION BRACKET~~
~~2E05 RIGHT DOTTED SUBSTITUTION BRACKET~~
~~2E09 LEFT TRANSPOSITION BRACKET~~
~~2E0A RIGHT TRANSPOSITION BRACKET~~
~~2E0C LEFT RAISED OMISSION BRACKET~~
~~2E0D RIGHT RAISED OMISSION BRACKET~~
~~2E1C LEFT LOW PARAPHRASE BRACKET~~
~~2E1D RIGHT LOW PARAPHRASE 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~~
~~301D REVERSED DOUBLE PRIME QUOTATION MARK~~
~~301E DOUBLE PRIME QUOTATION MARK~~
~~301F LOW DOUBLE PRIME QUOTATION MARK~~
~~FE59 SMALL LEFT PARENTHESIS~~
~~FE5A SMALL RIGHT PARENTHESIS~~
~~FE5B SMALL LEFT CURLY BRACKET~~
~~FE5C SMALL RIGHT CURLY BRACKET~~
~~FE5D SMALL LEFT TORTOISE SHELL BRACKET~~
~~FE5E SMALL RIGHT TORTOISE SHELL BRACKET~~
~~FE64 SMALL LESS-THAN SIGN~~
~~FE65 SMALL GREATER-THAN SIGN~~
~~FF08 FULLWIDTH LEFT PARENTHESIS~~
~~FF09 FULLWIDTH RIGHT PARENTHESIS~~
~~FF1C FULLWIDTH LESS-THAN SIGN~~
~~FF1E FULLWIDTH GREATER-THAN SIGN~~
~~FF3B FULLWIDTH LEFT SQUARE BRACKET~~
~~FF3D FULLWIDTH RIGHT SQUARE BRACKET~~
~~FF5B FULLWIDTH LEFT CURLY BRACKET~~
~~FF5D FULLWIDTH RIGHT CURLY BRACKET~~
~~FF5F FULLWIDTH LEFT WHITE PARENTHESIS~~
~~FF60 FULLWIDTH RIGHT WHITE PARENTHESIS~~
~~FF62 HALFWIDTH LEFT CORNER BRACKET~~
~~FF63 HALFWIDTH RIGHT CORNER BRACKET~~
~~1D6DB MATHEMATICAL BOLD PARTIAL DIFFERENTIAL~~
~~1D715 MATHEMATICAL ITALIC PARTIAL DIFFERENTIAL~~

~~1D74F MATHEMATICAL BOLD ITALIC PARTIAL DIFFERENTIAL~~
~~1D789 MATHEMATICAL SANS-SERIF BOLD PARTIAL DIFFERENTIAL~~
~~1D7C3 MATHEMATICAL SANS-SERIF BOLD ITALIC PARTIAL DIFFERENTIAL~~

Annex F
(informative)
Format characters

There is a special class of characters called Format characters the primary purpose of which is to affect the layout or processing of characters around them. With few exceptions, these characters do not have printable graphic symbols and, like the space characters, are represented in the character code tables by dotted boxes.

The function of most of these characters is to indicate the correct presentation of a CC-data element. 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~~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 are to be treated as a unit for the purpose of language-sensitive collation and searching. In language-sensitive collation and searching, the combining grapheme joiner should be ignored unless it specifically occurs with a tailored collation element mapping. For rendering, the combining grapheme joiner is invisible.~~

> ~~NOTE 1 – The combining grapheme joiner may be used to differentiate two usages of a combining character by using it for one of the two cases. For example, where a distinction is needed between the German umlaut and the tréma, the COMBINING GRAPHEME JOINER (034F) followed by the COMBINING DIAERESIS (0308) should be used to represent the tréma while the COMBINING DIAERESIS (0308) alone should be used to represent the German umlaut.~~

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 format character that indicates a preferred intra-word line-break opportunity. If the line is broken at that point, then whatever mechanism is appropriate for intra-word line-breaks should be invoked, just as if the line break had been triggered by another mechanism, such as a dictionary lookup. Depending on the language and the word, that may produce different visible results, such as:

- inserting a graphic symbol indicating the hyphenation and breaking the line after it,
- inserting a graphic symbol indicating the hyphenation, breaking the line after the symbol and changing spelling in the divided word parts,
- not showing any visible change and simply breaking the line at that point.

The inserted graphic symbol, if any, can take a wide variety of shapes, such as HYPHEN (2010), ARMENIAN HYPHEN (058A), MONGOLIAN TODO SOFT HYPHEN (1806), as appropriate for the situation.

When encoding text that includes explicit line breaking opportunities, including actual hyphenations, characters such as HYPHEN, ARMENIAN HYPHEN, and MONGOLIAN TODO SOFT HYPHEN may be used, depending on the language.

When a SOFT HYPHEN is inserted into a CC-data-element to encode a possible hyphenation point (for example: "tug{00AD}gumi"), the character representation remains otherwise unchanged. When encoding a CC-data-element that includes characters encoding hard line breaks, including actual hyphenations, the

character representation of the text sequence must reflect any changes due to hyphenation (for example: "tugg{2010}" / "gumi", where / represents the line break).

NOTE 2 – The notations {00AD} and {2010} indicate the inclusion of the corresponding code points: 00AD and 2010 into the CC-data-elements. The curly brackets “{}” are not part of the CC-data elements.

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.

WORD JOINER (2060) and **ZERO WIDTH NO-BREAK SPACE** (FEFF): These characters behave like a NO-BREAK SPACE in that they indicate the absence of word boundaries, but unlike NO-BREAK SPACE they have no presentational width. For example, these characters 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 3 – For additional usages of the ZERO WIDTH NO-BREAK SPACE 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 word stem. 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~~A.1.1 ~~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~~F.2.1 Symmetric swapping format characters

The following characters are used in conjunction with the class of left/right handed pairs of mirrored characters described in clause 15~~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 mirrored characters described in clause 15~~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 mirrored characters described in clause 15~~19~~ are interpreted and rendered as OPENING and CLOSING characters as specified in that clause.

~~F.2.3~~F.2.2 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~~F.2.3 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~~A.1.1 ~~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.2.6~~A.1.1 ~~Kharoshthi virama~~

~~**KHAROSHTHI VIRAMA** (10A3F): This character, which indicates the suppression of an inherent vowel, when followed by a consonant, causes a combined form consisting of two or more consonants. When not followed by another consonant, it causes the consonant which precedes it to be written as subscript to the left of the letter before it and is not displayed as a visible stroke or dot as VIRAMAs are in other scripts.~~

~~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~~A.1.1 ~~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 1 – 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. (D_1-D_2) or (D_1-D_2-D_3).~~

~~NOTE 2 – 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~~A.1.1 ~~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 D_1 on the left and D_2 on the right.~~

~~**IDEOGRAPHIC DESCRIPTION CHARACTER ABOVE TO BELOW** (2FF1): The IDS introduced by this character describes the abstract form of the ideograph with D_1 above D_2.~~

~~**IDEOGRAPHIC DESCRIPTION CHARACTER LEFT TO MIDDLE AND RIGHT** (2FF2): The IDS introduced by this character describes the abstract form of the ideograph with D_1 on the left of D_2, and D_2 on the left of D_3.~~

~~**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.~~

~~Table F.1: Properties of ideographic description characters~~

~~Character Name: IDEOGRAPHIC DESCRIPTION CHARACTER ...~~	~~no. of DCs~~	~~IDS Acronym and Syntax~~	~~Relative positions of DCs~~	~~Example of IDS~~	~~IDS example represents:~~
~~LEFT TO RIGHT~~	~~2~~	~~IDC-LTR D_1-D_2~~	D1 D2	~~⿰亻母~~	~~㑀~~
~~ABOVE TO BELOW~~	~~2~~	~~IDC-ATB D_1-D_2~~	D1 D2	~~⿱八天~~	~~关~~
~~LEFT TO MIDDLE AND RIGHT~~	~~3~~	~~IDC-LMR D_1-D_2-D_3~~	D1 D2 D3	~~⿲彳言亍~~	~~徜~~
~~ABOVE TO MIDDLE AND BELOW~~	~~3~~	~~IDC-AMB D_1-D_2-D_3~~	D1 D2 D3	~~⿳从从目~~	~~昝~~
~~FULL SURROUND~~	~~2~~	~~IDC-FSD D_1-D_2~~	D1 D2	~~⿴日莫~~	~~莫~~
~~SURROUND FROM ABOVE~~	~~2~~	~~IDC-SAV D_1-D_2~~	D1 D2	~~⿵門卞~~	~~閑~~
~~SURROUND FROM BELOW~~	~~2~~	~~IDC-SBL D_1-D_2~~	D2 D1	~~⿶凵士~~	~~出~~
~~SURROUND FROM LEFT~~	~~2~~	~~IDC-SLT D_1-D_2~~	D1 D2	~~⿷匚虎~~	~~虒~~
~~SURROUND FROM UPPER LEFT~~	~~2~~	~~IDC-SUL D_1-D_2~~	D1 D2	~~⿸广舞~~	~~廡~~
~~SURROUND FROM UPPER RIGHT~~	~~2~~	~~IDC-SUR D_1-D_2~~	D1 D2	~~⿹勹去~~	~~匆~~
~~SURROUND FROM LOWER LEFT~~	~~2~~	~~IDC-SLL D_1-D_2~~	D2 D1	~~⿺辶交~~	~~送~~
~~OVERLAID~~	~~2~~	~~IDC-OVL D_1-D_2~~	D1 D2 ~~*~~	~~⿻从工~~	~~巫~~

~~* NOTE – D_1 and D_2 overlap each other. This diagram does not imply that D_1 is on the top left corner and D_2 is on the bottom right corner.~~

~~F.4~~F.3 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~~F.4 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
0603 ARABIC SIGN SAFHA
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 5.0 (see Annex M~~Annex M~~ for referencing information), for ARABIC END OF AYAH.

~~F.6~~F.5 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 1D173 MUSICAL SYMBOL BEGIN BEAM and 1D174 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

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

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

F.6 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 (E0001) 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 (UTR#20), available from the Unicode web site (http://www.unicode.org/reports/), 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', code points from E0020 to E007E), as well as a language tag identification character: LANGUAGE TAG (E0001) and a cancel tag character: CANCEL TAG (E007F).

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.

F.6.1 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 encoded using tag characters. No other characters are valid for expressing the tag arguments.

F.6.2 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.

F.6.3 Cancelling 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.

F.6.4 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 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 UNIFIED IDEOGRAPHS EXTENSION C,

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 with the annotation between parentheses.

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: “AllnamesAll.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~~Integrated in main body text, see 10.

Annex I
(informative)
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.

I.1.1 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 1 – The above description implies that any IDS may be nested within another IDS.

Each IDC has four properties as summarized in table ~~F~~I.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. (D_1 D_2) or (D_1 D_2 D_3).

NOTE 2 – 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.

I.1.2 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 D_1 on the left and D_2 on the right.

IDEOGRAPHIC DESCRIPTION CHARACTER ABOVE TO BELOW (2FF1): The IDS introduced by this character describes the abstract form of the ideograph with D_1 above D_2.

IDEOGRAPHIC DESCRIPTION CHARACTER LEFT TO MIDDLE AND RIGHT (2FF2): The IDS introduced by this character describes the abstract form of the ideograph with D_1 on the left of D_2, and D_2 on the left of D_3.

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.

Table FI.1: Properties of ideographic description characters

Character Name: IDEOGRAPHIC DESCRIPTION CHARACTER ...	no. of DCs	IDS Acronym and Syntax	Relative positions of DCs	Example of IDS	IDS example represents:
LEFT TO RIGHT	2	IDC-LTR D_1 D_2	D1 D2	⿰ 亻 母	毋
ABOVE TO BELOW	2	IDC-ATB D_1 D_2	D1 D2	⿱ 八 天	关
LEFT TO MIDDLE AND RIGHT	3	IDC-LMR D_1 D_2 D_3	D1 D2 D3	⿲ 彳 言 亍	衍
ABOVE TO MIDDLE AND BELOW	3	IDC-AMB D_1 D_2 D_3	D1 D2 D3	⿳ 从 从 日	昏
FULL SURROUND	2	IDC-FSD D_1 D_2	D1 D2	⿴ 口 莫	圆
SURROUND FROM ABOVE	2	IDC-SAV D_1 D_2	D1 D2	⿵ 門 卞	閈
SURROUND FROM BELOW	2	IDC-SBL D_1 D_2	D2 D1	⿶ 凵 士	出
SURROUND FROM LEFT	2	IDC-SLT D_1 D_2	D1 D2	⿷ 匚 虎	匿
SURROUND FROM UPPER LEFT	2	IDC-SUL D_1 D_2	D1 D2	⿸ 广 舞	廡
SURROUND FROM UPPER RIGHT	2	IDC-SUR D_1 D_2	D1 D2	⿹ 勹 去	匆
SURROUND FROM LOWER LEFT	2	IDC-SLL D_1 D_2	D2 D1	⿺ 辶 交	这
OVERLAID	2	IDC-OVL D_1 D_2	D1 D2 *	⿻ 从 工	巫

* NOTE – D_1 and D_2 overlap each other. This diagram does not imply that D_1 is on the top left corner and D_2 is on the bottom right corner.

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.

~~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~~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 retransmit 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 12~~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 in ISO/IEC 10646 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

The clause 24~~28~~ of this standard specifies rules for name formation and name uniqueness. These rules are used in other information technology coded character set standards such as ISO/IEC 646, ISO/IEC 6937, ISO/IEC 8859, and ISO/IEC 10367. This annex provides additional guidelines for the creation of these entity names.

NOTE – These guidelines do not apply to the names of CJK Ideographs and Hangul syllables which are formed using rules specified in clause 24.6~~28.6~~ and 24.7~~28.7~~ respectively.

Guideline 1

The name of an entity wherever possible denotes its customary meaning (for example, the character name: PLUS SIGN or the block name: BENGALI).

Some entities, such as characters, may have a name describing shapes, not usage, (for example, the character name: UPWARDS ARROW).

The name on an entity is not intended to identify its properties or attributes, or to provide information on its linguistic characteristics, except as defined in guideline 4 below.

Guideline 2

An acronym consists of Latin capital letters A to Z and digits and is associated with a name.

Acronyms may be used in entity names where usage already exists and clarity requires it. For example, the names of control functions are coupled with an acronym.

EXAMPLES

Name:	Acronym
LOCKING-SHIFT TWO RIGHT	LS2R
SOFT HYPHEN	SHY
INTERNATIONAL PHONETIC ALPHABET	IPA

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

Guideline 3

Character names and named UCS Sequence Identifiers only include digits 0 to 9 if spelling out the name of the corresponding digits(s) would be inappropriate.

NOTE – As an example the name of the character at the code point value~~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.

Guideline 4

Character names and named UCS Sequence Identifiers 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 guidelines 9 to 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.

For character names, 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. The sequence may start with the marks above the letters taken in upwards sequence, and follow with the marks below the letters taken in downwards sequence, or the reverse (below/above).

For named UCS Sequence Identifiers, where the sequence comprises a base letter with multiple marks, the name describes the individual characters in the sequence in which they are encoded in the sequence.

EXAMPLES

Ộ	LATIN CAPITAL LETTER O WITH CIRCUMFLEX AND DOT BELOW
Ḉ	LATIN CAPITAL LETTER C WITH CEDILLA AND ACUTE
Ṵ	LATIN CAPITAL LETTER U WITH OGONEK AND ACUTE

Guideline 5

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

Guideline 6

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
І	CYRILLIC CAPITAL LETTER BYELORUSSIAN-UKRAINIAN I

Guideline 7

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

EXAMPLES

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

Guideline 8

Where possible, named UCS Sequence Identifiers are constructed by appending the names of the constituent elements together while eliding duplicate elements. Should this process result in a name that al-

ready exists, the name is modified suitably to guarantee uniqueness among character names and named UCS Sequence Identifiers. The words WITH and AND may be included for additional clarity when needed.

Guideline 9

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

Guideline 10

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

Guideline 11

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)

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 23~~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 códaithe 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, 2003. ISBN 0-321-18578-1

The Unicode Consortium *The Unicode Standard*, Version 5.0. Reading, MA: Addison-Wesley Developer's Press, 2007. ISBN 0-321-48091-0

The Unicode Consortium *Unicode Standard Annexes, UAX#9, The Unicode Bidirectional Algorithm, UAX#15 Unicode Normalization Forms, Version 4.0.0* 2003, and related Unicode Technical Reports, available at:
http://www.unicode.org/reports/

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 systems*. 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 CC-data-element content definition,
- the adopted encoding~~coded representation~~ form (~~4-octet or 2-octet~~UTF-8, UTF-16, or UTF-32).

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 ISO/IEC 10646-1 and ISO/IEC 10646-2. These two arcs should not be used.

The first such arc following a 10646 arc identifies the CC-data-element content definition, and is referred as ‘level-3 (3)’.

NOTE 2 – This version of the standard specifies a single definition for CC-data-element content. That definition was formerly known as implementation level 3 in previous editions of this standard

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 3 – 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 4 – As an example, the object identifier for the subset comprising the collections BASIC LATIN, LATIN-1 SUPPLEMENT, and MATHEMATICAL OPERATORS is:

{iso standard 10646 (0} level-3 (3) collections (1) 1 2 39}

ISO/IEC 8824 also specifies object descriptors corresponding to object identifier values. For unrestricted repertoire, the corresponding object descriptor is as follows:

3 0 : "ISO 10646 level-3 unrestricted"

For a single collection with collection name "xxx".

3 1 : "ISO 10646 level-3 xxx"

For a repertoire comprising more than one collection, numbered m1, m2, etc.

3 1 : "ISO 10646 level-3 collections m1, m2, m3, .. "

NOTE 5 – 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 encoding form and is either

~~two-octet-BMP-form (2), or~~
four-octet-form (4) for the UTF-32 encoding form, or
utf16-form (5) for the UTF-16 encoding form, or
utf8-form (8) for the UTF-8 encoding form.

NOTE 1 – As an example, the object identifier for the ~~two-octet coded representation~~UTF-32 encoding form is:

{iso standard 10646 (0) transfer-syntaxes (0) ~~two~~four-octet-~~BMP-~~form (~~2~~4)}

The following ~~form~~ object identifier is also valid but deprecated:

{iso standard 10646 (1) transfer-syntaxes (0) ~~two~~four-octet-~~BMP-~~form (~~2~~4)}

NOTE 2 – Previous versions of this standard supported a two-octet-BMP-form (2) arc which is now deprecated.

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 30~~34~~ 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~~ point~~in the two-octet form~~, followed by the related additional information. Entries are arranged in ascending sequence of code ~~position~~point.

~~When an entry for a character is included in this annex an * symbol appears immediately following its name in the corresponding table in clause 34.~~**~~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 word-stem. This allows for the normal rules of Mongolian character shaping to apply, while indicating that there is no word boundary at that position.~~

~~**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.~~

~~**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.~~

~~**Kharoshthi virama**~~

~~**KHAROSHTHI VIRAMA** (10A3F): This character, which indicates the suppression of an inherent vowel, when followed by a consonant, causes a combined form consisting of two or more consonants. When not followed by another consonant, it causes the consonant which precedes it to be written as subscript to the left of the letter before it and is not displayed as a visible stroke or dot as VIRAMAs are in other scripts.~~

000E <control> (shift-out)

This control character is named SHIFT-OUT in 7 –bit environment and LOCKING-SHIFT ONE in 8-bit environment

000F <control> (shift-in)

This control character is named SHIFT-IN in 7 –bit environment and LOCKING-SHIFT ZERO in 8-bit environment

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 15~~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 15~~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 ISO/IEC 10646-1 the name of this character was:

LATIN CAPITAL LIGATURE AE

00E6 LATIN SMALL LETTER AE (ash)

In the first edition of ISO/IEC 10646-1 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 ISO/IEC 10646-1 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 ISO/IEC 10646-1 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 ISO/IEC 10646-1 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 ISO/IEC 10646-1 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 zinorit.

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.

05B8 HEBREW POINT QAMATS

This character may be used generically or as qamats gadol in orthography which distinguishes it from 05C7 HEBREW POINTS QAMATS QATAN.

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~~ code point 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~~ points 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 allocated in code ~~positions~~ points 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>.

180E MONGOLIAN VOWEL SEPARATOR

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.

~~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>.~~

1DA6 MODIFIER LETTER SMALL CAPITAL I

This character should not be used for UPA (Uralic Phonetic Alphabet) purpose, the character 1D35 MODIFIER LETTER CAPITAL I should be used instead.

1DAB MODIFIER LETTER SMALL CAPITAL L

This character should not be used for UPA (Uralic Phonetic Alphabet) purpose, the character 1D38 MODIFIER LETTER CAPITAL L should be used instead.

1DB0 MODIFIER LETTER SMALL CAPITAL N

This character should not be used for UPA (Uralic Phonetic Alphabet) purpose, the character 1D3A MODIFIER LETTER CAPITAL N should be used instead.

1DB8 MODIFIER LETTER SMALL CAPITAL U

This character should not be used for UPA (Uralic Phonetic Alphabet) purpose, the character 1D41 MODIFIER LETTER CAPITAL U should be used instead.

202F NARROW NO-BREAK-SPACE

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 word-stem. This allows for the normal rules of Mongolian character shaping to apply, while indicating that there is no word boundary at that position.

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.

3164 HANGUL FILLER

This character represents the fill value used with the standard spacing Jamos.

~~2361 APL FUNCTIONAL SYMBOL UP TACK DIAERESIS~~

~~Information for the character at 234A applies.~~

9FB9 CJK UNIFIED IDEOGRAPH-9FB9
9FBA CJK UNIFIED IDEOGRAPH-9FBA
9FBB CJK UNIFIED IDEOGRAPH-9FBB

These three characters are intended to represent a component at a specific position of a full ideograph. The ideographs representing the same structure without a preferred positional preference are encoded at 20509, 2099D, and 470C respectively.

FA1F CJK COMPATIBILITY IDEOGRAPH-FA1F

This character should be considered as an extension to the block of characters CJK UNIFIED IDEOGRAPHS EXTENSION A (see 23~~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 23~~27~~, is:

C	J	K	V
G - Hanzi - T	Kanji	Hanja	ChuNom
	藹		
	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 23~~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 23~~27~~, are:

C	J	K	V
G - Hanzi - T	Kanji	Hanja	ChuNom
	赴		
	A-2728		
	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.

~~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

10A3F KHAROSHTHI VIRAMA

This character, which indicates the suppression of an inherent vowel, when followed by a consonant, causes a combined form consisting of two or more consonants. When not followed by another consonant, it causes the consonant which precedes it to be written as subscript to the left of the letter before it and is not displayed as a visible stroke or dot as VIRAMAs are in other scripts.

1D13A MUSICAL SYMBOL MULTIREST

This symbol is used as a rest corresponding in length to a breve note, which is usually called double whole rest in American usage or breve rest in British usage. The character 1D129 MUSICAL SYMBOL MULTIPLE MEASURE REST can be used to represent rests of arbitrary lengths.

1D300 MONOGRAM FOR EARTH,
1D301 DIGRAM FOR HEAVENLY EARTH,
1D302 DIGRAM FOR HUMAN EARTH,
1D303 DIGRAM FOR EARTHLY HEAVEN,
1D304 DIGRAM FOR EARTHLY HUMAN,
1D305 DIGRAM FOR EARTH

A Tai Xuan Jing symbol comprises a combination of three elements: tian, di and ren, and these three Chinese words usually translate to heaven, earth and human, respectively. The character names of the six Tai Xuan Jing sym-bols in this International Standard, however, are based on an uncommon mapping; tian for heaven, di for human, and ren for earth. Users are advised to identify these symbols by their representative glyphs or Chinese annotations but not character names.

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 ISO/IEC 10646-1 and their amended code positions as now specified in this edition of ISO/IEC 10646.~~

~~In the First Edition of ISO/IEC 10646-1 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 5-lines header, as many lines as Hangul syllables specified in the First Edition of ISO/IEC 10646-1; each containing the following information organized in fixed width fields:~~

- ~~01-05 octet: First Edition of ISO/IEC 10646-1 code positions for Hangul syllables (`hhhh`)~~
- ~~05 octet: SEMICOLON ';' used as a separator~~
- ~~06-09 octet: Current Edition of ISO/IEC 10646 code positions for Hangul syllables (`hhhh`).~~

~~The format definition uses `'h'` as a hexadecimal unit.~~

~~Click on this highlighted text to access the cross-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~~information concerning mapping between Hangul syllables (and code points) that were specified in the first edition of ISO/IEC 10646-1 and their amended code .points is available in previous editions of this standard.

Annex R
(informative)
Names of Hangul syllables

This annex provides the full name and annotation~~names~~ of Hangul syllables ~~in two formats, both available~~ through a linked file~~s~~:

~~1) Tabular arrangement showing the syllable-name of each character in the block HANGUL SYLLABLES (AC00 - D7A3). The syllable-name is the final component of the full character name, and is derived as described in clause 28.7, steps 1 to 5, which is the definitive specification of the names in that block. The leftmost column of the table shows the cell numbers (00 - FF) of the corresponding characters. The headings of the other columns of the table show the row numbers of the characters.~~

~~NOTE 1 – The content linked to is a PDF file, using a format similar to this standard containing the tabular arrangement.~~

~~Click on this highlighted text to access the file containing the Hangul syllable names in tabular arrangement.~~

~~The content is also available as a separate viewable file in the same directory as this document. The file is named: "HangulTb.pdf".~~

~~2) The full name and annotation of the Hangul syllables are also provided in a machine-readable format that is accessible as a link to this document.~~

NOTE 2 – 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 5-lines header, as all the Hangul syllables, each line specified as follows:

- 01-04 octet: ~~UCS-2~~ code ~~position~~ point in hexadecimal notation,
- 05 octet: SPACE character,
- 06 octet until end of line: Hangul syllable with the annotation between parentheses.

Click on this highlighted text to access the file containing the Hangul syllable names.

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

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 30~~34~~. 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~~ points to which they are assigned.

The source references for CJK unified ideographs are specified in 23.1~~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~~ point 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.

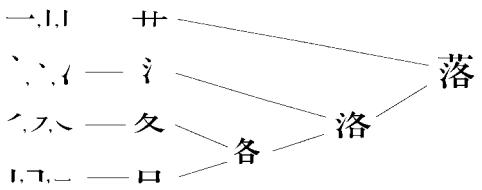


Figure S.1 - Component structure

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.

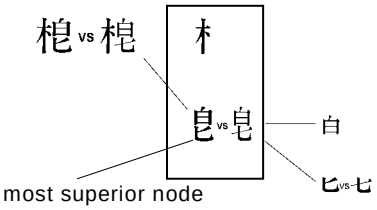


Figure S.2 - The most superior node of a component

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~~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•厷, 降•夅

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.

拡•擴, 策•筴, 学•学, 圣•圐, 奐•奐, 区•區, 夾•夾,
単•單, 雀•雚, 戔•戔 賛•贊, 襄•襄, 戴•戴, 間•閒,
朶•朵, 雋•隽, 恒•恒, 奐•奐, 从•从, 秦•秦, 夔•夔

S.1.5 Differences of actual shapes

To illustrate the classification described in S.1.2~~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.

辶•辶•辶, 示•示•礻, 旦•旦•旦, 食•食•食, 黄•黄, 盈•盈, 曷•曷,
包•包, 青•青, 每•每, 冊•冊, 爭•争, 謠•謠•謠, 彔•录,
歩•歩, 者•者, 臭•臭, 幷•并, 骨•骨, 呂•吕, 直•直,
県•県, 吴•吳•吳, 眞•真•真, 爲•為, 単•单, 曾•曾•曾, 成•成,
專•專, 内•内, 晉•晋, 龜•龜, 艹•艹

The differences are further classified according to the following examples.

a) Differences in rotated strokes/dots

半•半, 勺•勺, 羽•羽, 酉•酉,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

c) Differences in contact of strokes

奥•奥, 酉•酉, 査•査

d) Differences in protrusion at the folded corner of strokes

巨•巨

e) Differences in bent strokes

冊•冊

f) Differences in folding back at the stroke termination

奈•奈

g) Differences in accent at the stroke initiation

父•父, 丈•丈

h) Differences in "rooftop" modification

八•八, 穴•穴

i) Combinations of the above differences

刃•刃•刃

These differences in actual shapes of a unified ideograph are presented in the corresponding source columns for each code ~~position~~ point entry in the code charts in clause 30~~34~~ 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

NOTE – 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 charts of clause 30~~34~~ 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	Daejaweon 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 Dae-jaweon 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 S.1~~S.1~~. They are not unified because of the source separation rule described in S.1.6~~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	俱 俱	T	凈 淨	G	剏 剏	T
4E1F 4E22		4FF1 5036		51C0 51C8		524F 5259	
么 幺	GT	値 值	T	凡 凡	T	剝 剥	T
4E48 5E7A		5024 503C		51E2 51E3		525D 5265	
争 爭	GTJ	偷 偷	T	刃 刃	TJ	劒 劍	J
4E89 722D		5077 5078		5203 5204		5292 5294	
仞 仭	J	偽 僞	TJ	刊 刋	TJ	勻 勻	T
4EDE 4EED		507D 50DE		520A 520B		52FB 5300	
併 併	T	兌 兑	T	删 刪	T	单 单	T
4F75 5002		514C 5151		5220 522A		5355 5358	
侶 侶	T	兎 兔	TJ	别 别	T	即 卽	TK
4FA3 4FB6		514E 5154		5225 522B		5373 537D	
俣 俣	TJK	宛 宛	T	券 券	TJ	卷 卷	TJ
4FC1 4FE3		5156 5157		5238 52B5		5377 5DFB	
俞 俞	T	冊 冊	TJ	刹 刹	T	叁 参	GT
4FDE 516A		518A 518C		5239 524E		53C1 53C2	

參 叅	T	圖 圖	T	妍 妍	T	寧 寧	T
53C3 53C4		5716 5717		598D 59F8		5BDC 5BE7	
呂 呂	T	巠 巠	T	姍 姍	T	寢 寢	GTJ
5415 5442		5759 5DE0		59CD 59D7		5BDD 5BE2	
呑 呑	T	埒 埒	J	姬 姬	GT	專 專	J
541E 5451		57D2 57D3		59EB 59EC		5C02 5C08	
吳 吳 吳	TJ	堅 堅	T	娛 娛 娛	T	將 將	GTJ
5433 5434 5449		5848 588D		5A1B 5A2F 5A31		5C06 5C07	
吶 吶	T	塡 填	TJ	婕 婕	T	尒 尒	T
5436 5450		5861 586B		5A55 5AAB		5C13 5C14	
告 告	T	增 增	T	媮 媮	T	尚 尚	T
543F 544A		5897 589E		5A7E 5AAE		5C19 5C1A	
唧 唧	T	壯 壯	GTJ	媼 媼	TK	尪 尪	T
5527 559E		58EE 58EF		5AAA 5ABC		5C2A 5C2B	
喻 喻	T	壽 壽	T	媯 媯	T	尷 尷	T
55A9 55BB		58FD 5900		5AAF 5B00		5C36 5C37	
噓 噓	T	夐 夐	T	嫵 嫵	T	屏 屏	T
5618 5653		5910 657B		5B0E 5B14		5C4F 5C5B	
嚏 嚏	GTJ	本 本	GTJ	嬤 嬤	GT	崢 崢	GT
568F 5694		5932 672C		5B24 5B37		5CE5 5D22	
国 国	T	奧 奧	J	孿 孿	T	巔 巔	T
56EF 56FD		5965 5967		5B73 5B76		5DD3 5DD4	
圈 圈	TJ	奬 奬 奬	TJ	宮 宮	T	帡 帡	T
5708 570F		5968 596C 734E		5BAB 5BAE		5E21 5E32	
圓 圓	T	妝 妝	GT	寬 寬	T	帶 帶	TJ
570E 5713		5986 599D		5BDB 5BEC		5E2F 5E36	

并 并	T	惠 惠	TJ	挿 插 挿	TJ	曽 曾	J
5E76 5E77		6075 60E0		633F 63D2 63F7		66FD 66FE	
廏 廏	T	悦 悦	T	捏 揑	TJ	枴 枴	T
5EC4 5ECF		6085 60A6		634F 63D1		67B4 67FA	
弑 弑	T	悞 悞	T	捜 搜	TJ	査 査	T
5F11 5F12		609E 60AE		635C 641C		67E5 67FB	
強 强	T	悳 悳	T	掲 掲	T	柵 柵	T
5F37 5F3A		60B3 60EA		63B2 63ED		67F5 6805	
弾 弾	T	慍 慍	T	揺 搖 揺	TJ	梲 梲	T
5F39 5F3E		6120 614D		63FA 6416 6447		68B2 68C1	
彐 彑	TJ	愼 慎	TJ	揾 揾	T	楡 楡	T
5F50 5F51		613C 614E		63FE 6435		6961 6986	
彔 录	T	戩 戩	GT	撃 擊	TJ	概 概	T
5F54 5F55		6229 622C		6483 64CA		6982 69EA	
彙 彙	T	戲 戲	T	教 教	T	榲 榲	T
5F59 5F5A		622F 6231		654E 6559		6985 69B2	
彜 彝	J	戶 户 戸	T	敓 敓	T	樾 樾	T
5F5B 5F5C		6236 6237 6238		6553 655A		699D 6A27	
彝 彝	T	戻 戻	T	既 既	T	槇 槙	J
5F5D 5F5E		623B 623E		65E2 65E3		69C7 69D9	
彦 彦	T	抛 抛	T	昂 昂	T	樣 樣	TJ
5F65 5F66		629B 62CB		6602 663B		69D8 6A23	
徳 德	T	抜 拔	TJ	晩 晚	T	横 横	T
5FB3 5FB7		629C 62D4		665A 6669		6A2A 6A6B	
徵 徵	T	挩 挩	T	暨 暨	T	歩 歩	T
5FB4 5FB5		6329 635D		66A8 66C1		6B65 6B69	

歲 歲	T	淸 清	T	瑤 瑶	TJ	簨 簨	T
6B72 6B73		6DF8 6E05		7464 7476		7BB3 7C08	
歿 歿	T	渴 渴	T	甁 瓶	T	纂 纂	T
6B7F 6B81		6E07 6E34		74F6 7501		7BE1 7C12	
殼 殼	GTJ	温 溫	T	產 産	T	粵 粵	T
6BBB 6BBC		6E29 6EAB		7522 7523		7CA4 7CB5	
毀 毀	T	爲 爲	T	瘦 瘦	J	絕 絶	T
6BC0 6BC1		6E88 6F59		75E9 7626		7D55 7D76	
每 每	T	溉 漑	T	皐 皐	T	綠 緑	T
6BCE 6BCF		6E89 6F11		76A1 76A5		7DA0 7DD1	
氳 氳	T	滾 滾	T	眞 真	TJ	緒 緒	T
6C32 6C33		6EDA 6EFE		771E 771F		7DD2 7DD6	
汚 汚	T	潛 潛	GTJK	眾 衆	TJK	緣 緣	T
6C5A 6C61		6F5B 6FF3		773E 8846		7DE3 7E01	
沒 没	TJ	瀨 瀬	T	硏 研	T	縕 縕	T
6C92 6CA1		7028 702C		7814 784F		7DFC 7E15	
淨 淨	TJ	爲 爲	GTJ	祿 禄	TJ	繩 繩	T
6D44 6DE8		70BA 7232		797F 7984		7E48 7E66	
涉 涉	T	煢 煢	GTJK	禿 禿	T	羹 羹	TJ
6D89 6E09		712D 7162		79BF 79C3		7FAE 7FB9	
涗 涗	T	熙 熙	J	稅 税	T	翻 翻	T
6D97 6D9A		7155 7199		7A05 7A0E		7FF6 7FFA	
淚 淚	T	熅 熅	T	穗 穂	TJ	胼 胼	T
6D99 6DDA		7174 7185		7A42 7A57		80FC 8141	
淥 淥	T	狀 狀	GT	箏 箏	GJ	脫 脱	T
6DE5 6E0C		72B6 72C0		7B5D 7B8F		812B 8131	

腷 腷	T	蛻 蛻	T	达 述	T	閱 閱	T
817D 8183		86FB 8715		8FBE 8FD6		95B1 95B2	
舄 舄	GT	衛 衛	TJK	迸 迸	TJ	隉 隉	G
8203 8204		885B 885E		8FF8 902C		9667 9689	
舍 舍	TJ	衮 衮	TK	遙 遥	J	青 青	T
820D 820E		886E 889E		9059 9065		9751 9752	
舖 舖	J	装 装	GJK	邢 邢	T	静 靜	GTJ
8216 8217		88C5 88DD		90A2 90C9		9759 975C	
莊 莊	TJ	訝 訝	T	郎 郎	T	靱 靱	J
8358 838A		8A2E 8A7D		90CE 90DE		976D 9771	
蓿 蓿	TJ	說 說	T	鄉 鄉 鄉	T	頹 頹	T
83D1 8458		8AAA 8AAC		90F7 9109 9115		9839 983D	
葢 葢	T	諫 諫	TJ	醞 醞	T	顔 顔	TJ
8480 8495		8ACC 8AEB		9196 919E		984F 9854	
蒋 蔣	GJ	謠 謠	J	醬 醬	J	顛 顛	J
848B 8523		8B20 8B21		91A4 91AC		985A 985B	
蔿 蔿	T	豜 豜	T	鈃 鈃	T	飲 飲	J
848D 853F		8C5C 8C63		9203 9292		98EE 98F2	
蘊 蘊	T	走 走	TJ	鋭 鋭	T	餅 餅	TJ
8570 8580		8D70 8D71		92B3 92ED		9905 9920	
薫 薫	T	軿 軿	T	錄 録	T	馱 馱	TJK
85AB 85B0		8EFF 8F27		9304 9332		99B1 99C4	
蘊 蘊	T	輜 輜	J	錬 錬	TK	駢 駢	TK
85F4 860A		8F1C 8F3A		932C 934A		99E2 9A08	
虚 虚	T	輻 輻	T	鎮 鎮	TJ	骫 骫	T
865A 865B		8F3C 8F40		93AD 93AE		9AA9 9AAB	

高 高	T	鰛 鰛	TJ	鷆 鷆	J	黄 黄	T
9AD8 9AD9		9C1B 9C2E		9DC6 9DCF		9EC3 9EC4	
髮 髮	TJ	鳳 鳳	T	麪 麪	T	黑 黑	T
9AEA 9AEE		9CEF 9CF3		9EAA 9EAB		9ED1 9ED2	
鬪 鬪	T	鶇 鶇	J	麼 麼	T		
9B2C 9B2D		9D87 9DAB		9EBC 9EBD			

In accordance with the unification procedures described in S.1~~S.1~~ 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~~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~~S.1.6~~.

冑 冑	non cognate	寶 寶		朐 朐	non cognate	稻 稻	
5191 80C4		S.1.4.3~~S.1.4.3~~		6710 80CA		S.1.4.3~~S.1.4.3~~	
		5BF3 5BF6				7A32 7A3B	
沖 沖		廳 廳		朓 朓	non cognate	翱 翱	
S.1.4.3~~S.1.4.3~~		S.1.4.1~~S.1.4.1~~		6713 8101		S.1.4.3~~S.1.4.3~~	
51B2 6C96		5EF0 5EF3				7FF1 7FF6	
決 決		懷 懷		朘 朘	non cognate	耆 耆 耆	
S.1.4.3~~S.1.4.3~~		S.1.4.1~~S.1.4.1~~		6718 8127		S.1.4.3~~S.1.4.3~~	
51B3 6C7A		61D0 61F7				8007 8008 8009	
況 況		敥 敥		朣 朣	non cognate	聽 聽 聽	
S.1.4.3~~S.1.4.3~~		S.1.4.3~~S.1.4.3~~		6723 81A7		S.1.4.1~~S.1.4.1~~	
51B5 6CC1		6560 656A				8074 807C 807D	
垛 垛	S.1.4.3	肪 肪	non cognate	朵 朵		荆 荊	
579B 579C		670C 80A6		S.1.4.3~~S.1.4.3~~		S.1.4.2~~S.1.4.2~~	
				6735 6736		8346 834A	
孽 孽		朏 朏		灩 灩		躲 躲	
S.1.4.2~~S.1.4.2~~		non cognate		S.1.4.3~~S.1.4.3~~		S.1.4.3~~S.1.4.3~~	
5B7C 5B7D		670F 80D0		7054 7067		8EB1 8EB2	

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 (UTR#20), available from the Unicode web site (http://www.unicode.org/reports/), 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 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 Cancelling 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~~NOTE – Moved to F.6.

Annex U
(informative)
Characters in identifiers

A common task facing an implementer of UCS is the provision of a parsing and/or lexing engine for identifiers. Each programming language standard has its own identifier syntax; different programming languages have different conventions for the use of certain characters from the ASCII (ISO 646-IRV) range ($, @, #, _) in identifiers. Questions as to which characters to use for syntactic purposes versus which to be allowed in identifiers, whether case-pairing should be included, normalization should be performed, and other factors enter into the picture when defining the set of permitted characters for a given identification purpose.

Unicode Consortium publishes a document "UAX 31 – Identifier and Pattern Syntax" to assist in the standard treatment of identifiers in UCS character-based parsers. Those specifications are recommended for determining the list of UCS characters suitable for use in identifiers. The document is available at http://www.unicode.org/reports/tr31/.