ISO
INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION
--
ISO/IEC JTC1/SC2/WG2
Universal Multiple-Octet Coded Character Set (UCS)
--

ISO/IEC JTC1/SC2/WG2 ***N 1396***
Date: 1996-06-05

TITLE: ISO/IEC 10646-1 Corrigendum no. 2 (First draft - revised to 30 April 1996)
SOURCE: Bruce Paterson, project editor
STATUS: Working paper of JTC1/SC2/WG2
ACTION: For review and confirmation by WG2
DISTRIBUTION: Members of JTC1/SC2/WG2

1. Scope

This paper provides a first draft of Corrigendum no. 2 for ISO/IEC 10646-1, and replaces the previous paper WG2 N 1223R (1995-07-09).
It comprises editorial corrigenda for:
- the First Edition of ISO/IEC 10646-1, and
- Annex P of Technical Corrigendum no.1,

as agreed at WG2 meetings #24 to #30 (taken from WG2 N1207, N1254 and N1384).

This Corrigendum is provided in the form of complete replacement pages for:
- Clauses 1 to 26 (excluding figures and tables),
- Annexes A to D, F to N, and P,

and in the form of editing instructions for tables of graphic symbols and character names.
Editorial corrigenda for figures are not included (those for Figure 3 appear in DAM.5 - Hangul, and in WG2 N 1332).

In the replacement pages any changes that arise from Amendments nos. 1, 2, 3, and 4 have been applied to the text where applicable. However the new Annexes Q and R from Amendments nos. 1 and 2 are not attached to this WG2 paper since they do not form a part of the draft Corrigendum (they are available in N1334 and N1335 respectively).

2. Conventions used in the text

The typographical conventions used in this draft are as follows.
- Editing instructions are shown in italics.
- New text is shown underlined.
- Deleted text is retained in the draft, and is shown with strike-through style.

Note: If this Corrigendum is published by ITTF in the future, the underlining will be removed, and the deleted text will be absent, in accordance with the conventions for ITTF publications.

N1396 CR2

Draft Technical Corrigendum No.2 to ISO/IEC 10646-1: 1993 (E)

1996-06-05

NOTE: This draft Technical Corrigendum comprises the cumulative set of editorial corrigenda that have been approved by JTC1/SC2 for ISO/IEC 10646-1:1993 (E), from the date of publication of the First edition (1993-05-01) up to 1996-04-30. Some of the editorial corrigenda given here apply to Annex P that was added to the standard as part of Technical Corrigendum no. 1 (1996-03-01). Corrigenda applicable to Figure 3 (Overview of the Basic Multilingual Plane) appear in DAM5 (Hangul character collection) and are not repeated here.

This Technical Corrigendum replaces some of the pages of the First edition, and Annex P of Technical Corrigendum no.1, with revised pages in which the individual editorial corrigenda have already been applied to the text. In such pages the changes that arise from Amendments nos 1, 2, 3, and 4 have also been applied to the text where applicable. Other changes listed in Technical Corrigendum no. 1 are not repeated here.

1. *a) Replace pages 1 to 4 and 7 to 14 (text of clauses 1 to 25) with revised pages numbers 7 to 18 from Attachment 1 herewith.*

   b) Replace page 262 (text of clause 26) with revised page number 19 from Attachment 1 herewith.

2. *In clause 25 revise the titles of the following graphic symbol tables and character name tables as shown.*

   TABLE 6 - Row 02: IPA (INTERNATIONAL PHONETIC ALPHABET) EXTENSIONS *(pages 26 and 27)*
   TABLE 15 - Row 06: BASIC ARABIC *(pages 44 and 45)*
   TABLE 16 - Row 06: ARABIC EXTENDED *(pages 46 and 47)*
   TABLE 46 - Row 25: BLOCK ELEMENTS, GEOMETRIC SHAPES *(pages 106 and 107)*

3. *a) In Table 15 (Row 06: Arabic, page 44) remove the two right-hand columns of graphic symbols (numbered 066 and 067) and insert them joined to the left-hand side of Table 16 on page 46.*

   b) In Table 15 (Row 06: Arabic, page 45) delete all character name entries for code positions in the range hex 60 to hex 7F, and insert them into Table 16 (page 47) at the beginning of the list of character names (before code position hex 80).

4. *In clause 25, in the tables of graphic symbols identified in List EC.1 (see page 5 of this Corrigendum) replace the graphic symbols at the indicated code positions with the amended graphic symbols shown in the list.*

5. *In clause 25, Table 39 (Row 21: Arrows, page 92), transpose the graphic symbols at code positions 21DE and 21DF.*

6. *In clause 25, in the tables of character names indicated below, amend the entries listed below by inserting an * symbol after the final word as shown (each entry is identified by its code position number in the “hex” column).*

   Table 2 - Row 00 (page 19)

AB	LEFT-POINTING DOUBLE ANGLE QUOTATION MARK *
BB	RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK *

   Table 4 - Row 01 (page 23)

89	LATIN CAPITAL LETTER AFRICAN D *
9F	LATIN CAPITAL LETTER O WITH MIDDLE TILDE *

Table 16 - Row 06 (page 47)

AF ARABIC LETTER GAF *
D0 ARABIC LETTER E *

Table 42 - Row 23 (page 99)

4A APL FUNCTIONAL SYMBOL DOWN TACK UNDERBAR *
4E APL FUNCTIONAL SYMBOL DOWN TACK JOT *
51 APL FUNCTIONAL SYMBOL UP TACK OVERBAR *
55 APL FUNCTIONAL SYMBOL UP TACK JOT *
61 APL FUNCTIONAL SYMBOL UP TACK DIAERESIS *

Table 123 - Row FF (page 261)

E3 FULLWIDTH MACRON *

7. *In clause 25, in the tables of character names indicated below, amend the entries listed below by inserting text in parentheses as shown (each entry is identified by its code position number in the "hex" column).*

Table 4 - Row 01 (page 23)

A2 LATIN CAPITAL LETTER OI (gha)
A3 LATIN SMALL LETTER OI (gha)

Table 14 - Row 05 (page 43)

BC HEBREW POINT DAGESH OR MAPIQ (or shuruq)

Table 40 - Row 22 (page 95)

3D REVERSED TILDE (lazy S)

Table 123 - Row FF (page 261)

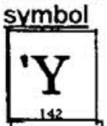
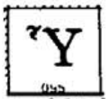
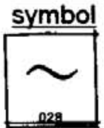
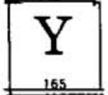
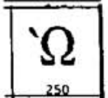
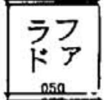
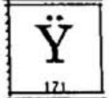
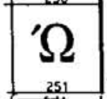
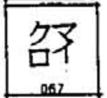
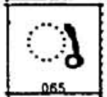
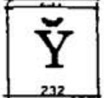
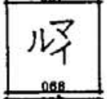
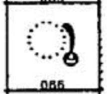
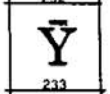
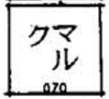
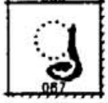
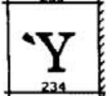
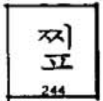
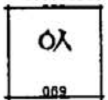
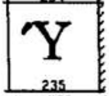
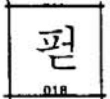
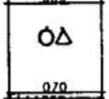
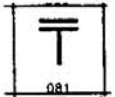
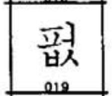
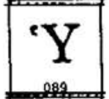
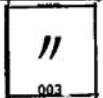
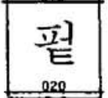
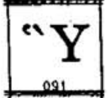
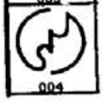
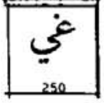
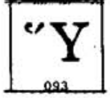
9E HALFWIDTH KATAKANA VOICED SOUND MARK
(halfwidth katakana-hiragana voiced sound mark)
9F HALFWIDTH KATAKANA SEMI-VOICED SOUND MARK
(halfwidth katakana-hiragana semi-voiced sound mark)

8. *In clause 26, in column J of the tables of graphic symbols on the pages identified in List EC.2 (see page 6 of this Corrigendum) replace the graphic symbols at the indicated code positions with the amended graphic symbols shown in the list.*

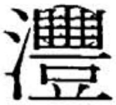
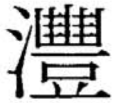
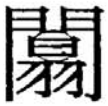
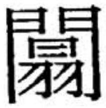
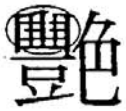
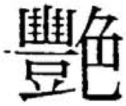
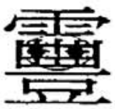
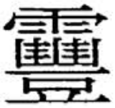
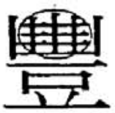
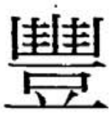
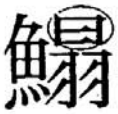
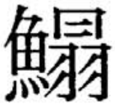
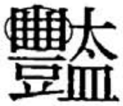
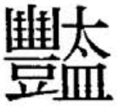
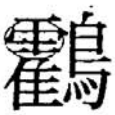
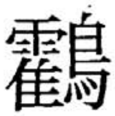
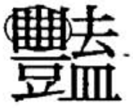
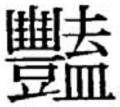
9. *Replace pages 699 to 708 (Annexes A to D) and 743 to 754 (Annexes F to N) with revised pages from Attachment 2 (pages 1 to 20) herewith.*

10. *Replace Annex P (of Technical Corrigendum no.1) with revised Annex P from Attachment 2 (page 21) herewith.*

List EC.1 - Amended symbols for Tables of graphic symbols (in clause 25)

Table	Code position	Graphic symbol	Table	Code position	Graphic symbol	Table	Code position	Graphic symbol
9	038E		33	1F5F		50	301C	
9	03A5		34	1FFA		56	3332	
9	03AB		34	1FFB		56	3343	
25	0D41		34	1FE8		56	3344	
25	0D42		34	1FE9		56	3346	
25	0D43		34	1FEA		87	42F4	
29	1145		34	1FEB		90	4412	
29	1146		42	2351		90	4413	
33	1F59		50	3003		90	4414	
33	1F5B		50	3004		116	FCFA	
33	1F5D							

List EC.2 - Amended symbols for column J of Tables of graphic symbols (in clause 26)

Page	Code position	replaced symbol	amended symbol	Page	Code position	replaced symbol	amended symbol
445	7043			646	95D2		
542	8277			653	974A		
595	8C50			679	9C28		
595	8C53			690	9E16		
595	8C54						

N1396-A

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

Part 1:

Architecture and Basic Multilingual Plane

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 additional symbols.
This part of ISO/IEC 10646 specifies the overall architecture, and

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

The UCS is a coding system different from that specified in ISO 2022. The method to designate UCS from ISO 2022 is specified in 17.2.

2 Conformance

2.1 General

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

2.2 Conformance of information interchange

A coded-character-data-element (CC-data-element) within coded information for interchange is in conformance with ISO/IEC 10646 if
a) all the coded representations of graphic characters within that CC-data-element conform to clauses 6 and 7, to an identified form chosen from clause 14 or Annex Q or Annex R, and to an identified implementation level chosen from clause 15;
b) all the graphic characters represented within that CC-data-element are taken from those within an identified subset (clause 13);
c) all the coded representations of control functions within that CC-data-element conform to clause 16.
A claim of conformance shall identify the adopted form, the adopted implementation level and the adopted subset by means of a list of collections and/or characters.

2.3 Conformance of devices

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

NOTE - The term device is defined (in 4.17) 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 applicationprogram or gateway function.

A claim of conformance shall identify the document that contains the description specified in a) below, and shall identify the adopted form(s), the adopted implementation level, the adopted subset (by means of a list of collections and/or characters), and the selection of control functions adopted in accordance with clause 16.
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 recognise 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 form and implementation level.
c) Receiving device: A receiving device shall be capable of receiving and interpreting any coded representation of characters that are within a CC-data-element in accordance with the adopted form and implementation level, and shall make any corresponding characters from the adopted subset available to the user in such a way that the user can identify them.
Any corresponding characters that are not within the adopted subset shall be indicated to the user in a way which need not allow them to be distinguished from each other.

NOTES

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.

2 See also annex H for receiving devices with re-transmission capability.

3 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 10646. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 10646 are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.
~~ISO 2022:1986 *Information processing ISO 7-bit and 8-bit coded character sets — Code extension techniques.*~~
ISO/IEC 2022:1994 *Information technology — Character code structure and extension techniques.*
ISO/IEC 6429:1992 *Information technology — Control functions for coded character sets.*

4 Definitions

For the purposes of ISO/IEC 10646, the following definitions apply :
4.1 Basic Multilingual Plane (BMP): Plane 00 of Group 00.
4.2 block: A contiguous collection of characters that share common characteristics, such as script.
4.3 canonical form: The form with which characters of this coded character set are specified using four octets to represent each character.
4.4 CC-data-element (Coded-Character-Data-Element): An element of interchanged information that is specified to consist of a sequence of coded representations of characters, in accordance with one or more identified standards for coded character sets.
4.5 cell: The place within a row at which an individual character may be allocated.
4.6 character: A member of a set of elements used for the organisation, control, or representation of data.
4.7 character boundary: Within a stream of octets the demarcation between the last octet of the coded representation of a character and the first octet of that of the next coded character.
4.8 coded character: A character together with its coded representation.
4.9 coded character set: A set of unambiguous rules that establishes a character set and the relationship between the characters of the set and their coded representation.
4.10 code table: A table showing the characters allocated to the octets in a code
4.11 collection: A set which is numbered and named and which consists of named characters taken from this standard.
4.12 combining character: A member of an identified subset of the coded character set of ISO/IEC 10646 intended for combination with the preceding non-combining graphic character, or with a sequence of combining characters preceded by a non-combining character (see also 4.13).

NOTE - This part of ISO/IEC 10646 specifies several subset collections which include combining characters.

4.13 compatibility character: A graphic character included as a coded character of ISO/IEC 10646 primarily for compatibility with existing coded character sets.
4.14 composite sequence: A sequence of graphic characters consisting of a non-combining character followed by one or more combining characters (see also 4.11).

NOTES

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

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

4.15 control function: An action that affects the recording, processing, transmission or interpretation of data, and that has a coded representation consisting of one or more octets.
4.16 default state: The state that is assumed when no state has been explicitly specified.
4.17 detailed code table: A code table showing the individual characters, and normally showing a partial row.
4.18 device: A component of information processing equipment which can transmit and/or receive coded information within CC-data-elements. (It may be an

input/output device in the conventional sense, or a process such as an application program or gateway function.)

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

4.20 graphic symbol: The visual representation of a graphic character or of a composite sequence.

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

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

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

4.24 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.

4.25 low-half zone: a set of cells reserved for use in UTF-16 (see Annex Q); an RC-element corresponding to any of these cells may be used as the second of a pair of RC-elements which represents a character from a plane other than the BMP.

4.26 octet: An ordered sequence of eight bits considered as a unit.

4.27 plane : A subdivision of a group; of 256 x 256 cells

4.28 presentation; to present: The process of writing, printing, or displaying a graphic symbol.

4.29 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.

4.30 private use planes: ~~Planes~~ A plane within this coded character set the contents of which ~~are~~ is not specified in ISO/IEC 10646 (see 10.1)

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

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

4.33 row: A subdivision of a plane; of 256 cells.

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

4.35 supplementary planes: ~~Planes~~ A plane that accommodates characters which have not been allocated to the Basic Multilingual Plane.

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

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

4.37 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.)

4.38 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 (see clause 8).

5 General structure of the UCS

The general structure of the Universal Multiple-Octet Coded Character Set (referred to hereafter as "this coded character set") is described in this explanatory clause, and is illustrated in figures 1 and 2. The normative specification of the structure is given in ~~later~~ the following clauses.

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

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

NOTE - 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.

The first plane (Plane 00 of Group 00) is called the Basic Multilingual Plane. The Basic Multilingual Plane includes characters in general use in alphabetic, syllabic and ideographic scripts together with various symbols and digits. The BMP also has a restricted use ~~(RU)~~ zone (R-zone) in which the characters have special characteristics (see clauses 8 and 10).

The subsequent planes are regarded as supplementary or private use planes, which will accommodate additional graphic characters (see clause 9).

The planes that are reserved for private use are specified in clause 11. ~~The 32 planes with Plane-octet values E0 to FF of Group 00 are for Private Use. The 32 groups with Group-octet values 60 to 7F of this coded character set are also for Private Use~~. The contents of the cells in private use 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.

In addition to the canonical form, a two-octet BMP form is specified. Thus, the Basic Multilingual Plane can be used as a two-octet coded character set identified as UCS-2.

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

~~A UCS Transformation Format (UTF-1) is specified in annex G which can be used to transmit text data through communication systems which are sensitive to octet values for control characters coded according to the structure of ISO 2022.~~

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

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

6 Basic structure and nomenclature

6.1 Structure

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

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

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

Accordingly, the weight allocated to each bit shall be

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

6.2 Coding of characters

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

m.s. l.s.

Group-octet	Plane-octet	Row-octet	Cell-octet

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

For brevity, the octets may be termed

m.s. l.s.

G-octet	P-octet	R-octet	C-octet

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

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

0000 0030 for DIGIT ZERO

0000 0041 for LATIN CAPITAL LETTER A

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

6.3 Octet order

The sequence of the octets that represent a character, and the most significant and least significant ends of it, shall be maintained as shown above. When serialised as octets, a more significant octet shall precede less significant octets. When not serialised as octets, the order of octets may be specified by agreement between sender and recipient (see 17.1 and annex F).

6.4 Naming of characters

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

a. denotes the customary meaning of the character, or

b. describes the shape of the corresponding graphic symbol, or

c. follows the rule given in clause 26 for Chinese/Japanese/Korean unified ideographs.

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

7 Special features of the UCS

The following characteristics apply to the entire coded character set.

a) ~~1.~~ The values of P-, and R-, and C-octets used for representing graphic characters shall be in the range 00 to FF. The values of G-octets used for representation of graphic characters shall be in the range 00 to 7F. On any plane, code positions FFFE and FFFF shall not be used.

NOTE - Code position FFFE is reserved for "signature" (see annex F). Code position FFFF can be used for internal processing uses requiring a numeric value that is guaranteed not to be a coded character such as in terminating tables, or signaling end-of-text. Since it is the largest two-octet value, it may also be used as the final value in binary or sequential searching index.

b) ~~2.~~ 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 standardisation and shall not be used for any other purpose. Future editions of ISO/IEC 10646 will not allocate any characters to code positions reserved for private use characters or for transformation formats.

c) ~~3.~~ 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.

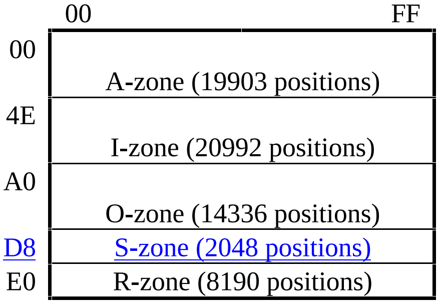
~~4. Compatibility characters are included in ISO/IEC 10646 primarily for compatibility with existing coded character sets to allow two-way code conversion without loss of information.~~

8 The Basic Multilingual Plane

Plane 00 of Group 00 shall be the Basic Multilingual Plane (BMP). The BMP can be used as a two-octet coded character set in which case it shall be called UCS-2 (see 14.1).

The Basic Multilingual Plane shall be divided into five ~~four~~ zones:

A-zone: code positions 0000 0000 to 0000 4DFF
I-zone: code positions 0000 4E00 to 0000 9FFF
O-zone: code positions 0000 A000 to 0000 D7FF
S-zone code positions 0000 D800 to 0000 DFFF
R-zone: code positions 0000 E000 to 0000 FFFD



Code positions 0000 0000 to 0000 001F in the BMP are reserved for control characters, and code position 0000 007F is reserved for the character DELETE (see clause 16). Code positions 0000 0080 to 0000 009F are reserved for control characters.

In the Basic Multilingual Plane, the A-zone is used for alphabetic and syllabic scripts together with various symbols. The I-zone is used for Chinese/Japanese/Korean (CJK) unified ideographs (unified East Asian ideographs). The O-zone is reserved for future standardisation. The S-zone is reserved for the use of UTF-16 (see Annex Q). The R-zone shall be used for the restricted use zone in the BMP which contains private use characters, presentation forms, and compatibility characters (see clause 10) .

9 Other planes

9.1 Planes reserved for future standardisation

Planes 11 ~~01~~ to DF in Group 00 and planes 00 to FF in Groups 01 to 5F are reserved for future standardisation, and thus those code positions shall not be used for any other purpose.

9.2 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 Q). This form is compatible with the two-octet BMP form of UCS-2 (see 14.1).

Code positions in planes 11 to FF of group 00, or in planes 00 to FF of other groups, do not have a mapping to the UTF-16 form.

10 The restricted use zone

Sets of graphic characters that are used in particular ways are provided in the restricted use zone. These sets include:

a) Private use characters,
b) Presentation forms of characters,
c) Compatibility characters ~~(see item 4 in clause 7)~~.

10.1 Private use characters

Private use characters are not restrained 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 characters ~~(DRCS)~~ applications.

NOTE 2 - For meaningful interchange of ~~DRCS~~ 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 Presentation forms of characters

Each presentation form of 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.

10.3 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.

11 Private use groups, planes and zones

The code positions of the 32 groups from Group 60 to Group 7F shall be for private use.

The code positions of Plane 0F and Plane 10, and of the 32 planes from Plane E0 to Plane FF, 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).

12 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.

13 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.

13.1 Limited subset

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

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

13.2 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 of each part of ISO/IEC 10646. A selected subset shall always automatically include the Cells 20 to 7E of Row 00 of Plane 00 of Group 00.

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

14 Coded representation forms of the UCS

ISO/IEC 10646 provides two alternative forms of coded representation of characters.

> 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.

14.1 Two-octet BMP form

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

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

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

14.2 Four-octet canonical form

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

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

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

15 Implementation levels

ISO/IEC 10646 specifies three levels of implementation. Combining characters are described in 23 and listed in annex B.

15.1 Implementation level 1

When implementation level 1 is used, a CC-data-element shall not contain coded representations of combining characters (see clause B.1) nor of characters from HANGUL JAMO block (see clause 24).

15.2 Implementation level 2

When implementation level 2 is used, a CC-data-element shall not contain coded representations of characters listed in clause B.2.

15.3 Implementation level 3

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

16 Use of control functions with the UCS

This coded character set provides for use of control functions encoded according to ~~ISO 2022,~~ 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 ~~C0~~ control character of ISO/IEC 6429 is used with this coded character set, its coded representation as specified in ISO/IEC 6429 shall be padded to correspond with the number of octets in the adopted form (see clause 14). Thus, the least significant octet shall be the bit combination specified in ISO/IEC 6429, and the more significant octet(s) shall be zeros.

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

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

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

~~When using a C1 control character of ISO/IEC 6429 with this coded character set, it shall be coded as ESC Fe sequence (see ISO/IEC 6429) padded as specified above.~~

~~For example, the control character PARTIAL LINE BACKWARD - PLU (08/12 in ISO/IEC 6429 representation) is represented by "001B 004C" in the two-octet form, and "0000 001B 0000 004C" in the four-octet form.~~

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

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

17 Declaration of identification of features

17.1 Purpose and context of identification

CC-data-elements conforming to ISO/IEC 10646 are intended to form all or part of a composite unit of coded information that is interchanged between an originator and a recipient. The identification of ISO/IEC 10646 (including the form), the implementation level, and any subset of the coding space that have been adopted by the originator must also be available to the recipient. The route by which such identification is communicated to the recipient is outside the scope of ISO/IEC 10646. However, some standards for interchange of coded information may permit, or require, that the coded representation of the identification applicable to the CC-data-element forms a part of the interchanged information. This clause specifies a coded representation for the identification of UCS with an implementation level and a subset of ISO/IEC 10646, and also of a C0 and a C1 set of control functions from ISO/IEC 6429 for use in conjunction with ISO/IEC 10646. Such coded representations provide all or part of an identification data element, which may be included in information interchange in accordance with the relevant standard.

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

NOTE - An alternative method of identification is described in annex M.

17.2 Identification of UCS coded representation form with implementation level

When the escape sequences from ISO/IEC 2022 are used, the identification of a coded representation form of UCS (see clause 14) and an implementation level (see clause

15) specified by ISO/IEC 10646 shall be by a designation sequence chosen from the following list:

ESC 02/05 02/15 04/00
UCS-2 with implementation level 1
ESC 02/05 02/15 04/01
UCS-4 with implementation level 1
ESC 02/05 02/15 04/03
UCS-2 with implementation level 2
ESC 02/05 02/15 04/04
UCS-4 with implementation level 2
ESC 02/05 02/15 04/05
UCS-2 with implementation level 3
ESC 02/05 02/15 04/06
UCS-4 with implementation level 3

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

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

17.3 Identification of subsets of graphic characters

When the control sequences of ISO/IEC 6429 are used, the identification of subsets (see clause 13) specified by ISO/IEC 10646 shall be by a control sequence IDENTIFY UNIVERSAL CHARACTER SUBSET (IUCS) as shown below.

CSI Ps... 02/00 06/13

Ps... means that there can be any number of selective parameters. The parameters are to be taken from the subset collection numbers as shown in annex A of each part 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 16.

17.4 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 16) of ISO/IEC 6429 to be used in conjunction with ISO/IEC 10646 shall be an identifier sequence of the type shown below.

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

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

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

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

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

~~17.5 Identification of return from UCS to ISO 2022~~

~~When the escape sequences from ISO 2022 are used, the identification of the return from UCS to the coding system of ISO 2022 shall be by the escape sequence ESC 02/05 04/00, padded in accordance with clause 16.~~

17.5 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 16.

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.

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 17.2 for the identification of UCS include the octet 02/15 to indicate that the return does not always conform to that standard.

18 Structure of the code tables and lists

The clauses 25 and 26 set out the detailed code tables and the lists of character names for the graphic characters. Together, these specify graphic characters, their coded representation, and the character name for each character.

The graphic symbols are to be regarded as typical visual representations of the characters. ISO/IEC 10646 does not attempt to prescribe the exact shape of each character. The shape is affected by the design of the font employed, which is outside the scope of ISO/IEC 10646. Graphic characters specified in ISO/IEC 10646 are uniquely identified by their names. This does not imply that the graphic symbols by which they are commonly imaged are always different. Examples of graphic characters with similar graphic symbols are LATIN CAPITAL LETTER A, GREEK CAPITAL LETTER ALPHA, and CYRILLIC CAPITAL LETTER A.

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

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

It is not intended, nor will it often be the case, that the characters needed by any one user will be found all grouped together in one part of the code table. Furthermore, the user of any script will find that needed characters ~~he needs~~ may have been ~~already~~ coded ~~earlier~~ 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 clause 19 or an overview of the BMP in figure 3, and then to turn to the specific code table rows for the relevant script and for symbols and digits. In addition, annex E contains an alphabetically sorted list of character names.

19 Block names

The following list contains the blocks defined in the BMP. The block names are used in providing for the specification of subsets (see annex A for subset collections).

Block name	from	to
BASIC LATIN	0020	007E
LATIN-1 SUPPLEMENT	00A0	00FF
LATIN EXTENDED-A	0100	017F
LATIN EXTENDED-B	0180	024F
IPA EXTENSIONS	0250	02AF
SPACING MODIFIER LETTERS	02B0	02FF
COMBINING DIACRITICAL MARKS	0300	036F
BASIC GREEK	0370	03CF
GREEK SYMBOLS AND COPTIC	03D0	03FF
CYRILLIC	0400	04FF
ARMENIAN	0530	058F
HEBREW EXTENDED-A	0590	05CF
BASIC HEBREW	05D0	05EA
HEBREW EXTENDED-B	05EB	05FF
~~BASIC ARABIC~~	~~0600~~	~~0652~~
BASIC ARABIC	0600	065F
~~ARABIC EXTENDED~~	~~0653~~	~~06FF~~
ARABIC EXTENDED	0660	06FF
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
THAI	0E00	0E7F
LAO	0E80	0EFF
GEORGIAN EXTENDED	10A0	10CF
BASIC GEORGIAN	10D0	10FF
HANGUL JAMO	1100	11FF
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
CJK SYMBOLS AND PUNCTUATION	3000	303F
HIRAGANA	3040	309F
KATAKANA	30A0	30FF
BOPOMOFO	3100	312F
HANGUL COMPATIBILITY JAMO	3130	318F
CJK MISCELLANEOUS	3190	319F
ENCLOSED CJK LETTERS AND MONTHS	3200	32FF
CJK COMPATIBILITY	3300	33FF
HANGUL	3400	3D2D
HANGUL SUPPLEMENTARY-A	3D2E	44B7
HANGUL SUPPLEMENTARY-B	44B8	4DFF
CJK UNIFIED IDEOGRAPHS	4E00	9FFF
PRIVATE USE AREA	E000	F8FF
CJK COMPATIBILITY IDEOGRAPHS	F900	FAFF
ALPHABETIC PRESENTATION FORMS	FB00	FB4F
ARABIC PRESENTATION FORMS-A	FB50	FDFF
COMBINING HALF MARKS	FE20	FE2F
CJK COMPATIBILITY FORMS	FE30	FE4F
SMALL FORM VARIANTS	FE50	FE6F
ARABIC PRESENTATION FORMS-B	FE70	FEFE
HALFWIDTH AND FULLWIDTH FORMS	FF00	FFEF
SPECIALS	FFF0	FFFD

20 Characters in bi-directional context

A class of left/right handed pairs of characters have special significance in the context of bi-directional text.

In this context the terms LEFT or RIGHT in the character name are also intended to imply "opening" or "closing" forms of character shape, rather than a strict left-hand or right-hand form. These characters are listed below.

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

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

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

> NOTE - In the context of Arabic bi-directional text, a set of mathematical symbols may also have special significance (see annex C).

21 Special characters

There are some characters that do not have printable graphic symbols. These characters include space characters. They are

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

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.

There is a special class of characters called Alternate Format Characters which are included for compatibility with some industry practices. These are described in annex D.

22 Order of characters

Usually, coded characters appear in a CC-data-element in logical order (logical or backing store order corresponds to the order in which characters are entered from the keyboardafter 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 the short i in Devanagari is displayed before the character that it logically follows in the CC-data-element.

23 Combining characters

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

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

23.1 Order of combining characters

Coded representations of combining characters shall follow that of the graphic character with which they are associated (for example, coded representations of LATIN SMALL LETTER A followed by COMBINING TILDE represent a composite sequence for Latin "ã").

If a combining character is to be regarded as a composite sequence in its own right, it shall be coded as a composite sequence by association with the character SPACE. For example, grave accent can be composed as SPACE followed by COMBINING GRAVE ACCENT.

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

23.2 Appearance in code tables

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

In the code tables for some scripts, such as Hebrew, Arabic, and the scripts of India and South-east Asia, combining characters are indicated in relation to dotted circles to show their position relative to the base character. Many of these combining characters encode vowel letters; as such they are not generally referred to as "diacritical marks".

23.3 Multiple combining characters

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

a) ~~1.~~ If the combining characters can interact in presentation (for example, a combining macron and a combining diaeresis), then the position of the combining characters in the resulting graphic display is determined by the order of the coded representation of the combining characters. The presentations of combining characters are to be positioned from the base character outward. For example, combining characters placed above a base character are stacked vertically, starting with the first encountered in the sequence of coded representations and continuing for as many marks above as are required by the coded combining characters following the coded base character. For combining characters placed below a base character, the situation is inverted, with the combining characters starting from the base character and stacking downward.

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

b) ~~2.~~ 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 (0000 0343) requires that, together with a following acute or grave accent, they be rendered side-by-side above a letter, rather than the accent marks being stacked above the COMBINING GREEK KORONIS. The order of the coded representations is: the letter itself, followed by that of the breathing mark, followed by that of the accent marks. Two Vietnamese tone marks which have the same graphic appearance as the Latin acute and grave accent marks do not stack above the three Vietnamese vowel letters which already contain the circumflex diacritic (â, ê, ô). Instead, they form ligatures with the circumflex component of the vowel letters.

c) ~~3.~~ 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.

NOTE - Where combining characters are used for the generation of composite sequences in implementation level 3, this facility may be used to provide an alternative coded representation of text. For example, in implementation level 3 the French word "là" may be represented by the characters LATIN SMALL LETTER L followed by LATIN SMALL LETTER A WITH GRAVE, or may be represented by the characters LATIN SMALL LETTER L followed by LATIN SMALL LETTER A followed by COMBINING GRAVE ACCENT.

23.4 Collections containing combining characters

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

When implementation level 1 or 2 is adopted, a CC-data-element shall not contain the coded representations of combining characters listed in annex B, even though the adopted subset may include them.
Other collections of characters listed in annex A comprise only combining characters, for example collection 7 (COMBINING DIACRITICAL MARKS). Such a collection shall not be included in the adopted subset when implementation level 1 is adopted.

24 Hangul syllable composition method

In rendering, a sequence of Hangul Jamo (from HANGUL JAMO block: 1100 to 11FF) are displayed as a series of syllable blocks. Jamo can be classified into three classes: Choseong (syllable-initial character), Jungseong (syllable-peak character), and Jongseong (syllable-final character). A complete syllable block is composed of a Choseong and a Jungseong, and optionally a Jongseong. An incomplete syllable is a string of one or more characters which does not constitute a complete syllable (for example, a Choseong alone, a Jungseong alone, a Jongseong alone, or a Jungseong followed by a Jongseong). An incomplete syllable which starts with a Jungseong or a Jongseong must be preceded by a CHOSEONG FILLER (0000 115F). An incomplete syllable composed of a Choseong alone must be followed by a JUNGSEONG FILLER (0000 1160).
~~Hangul Jamo are conjoining characters since they do not require non-combining characters for the syllable composition method.~~ The implementation level 3 shall be used for the Hangul syllable composition method.

NOTE - Hangul Jamo are not combining characters.

25 Code tables and lists of character names

An overview of the Basic Multilingual Plane is shown in figure 3. Detailed code tables and lists of character names for the Basic Multilingual Plane are shown on the following pages.
Guidelines to be used for constructing names of characters are given in annex K for information. In some cases, a name of a character is followed by additional explanatory statements not part of the name. These statements are in parentheses and not in capital letters except for the initials of the word, where required.

26 CJK unified ideographs

Entries in the code tables for CJK (Chinese/Japanese/Korean) unified ideographs are arranged as follows:

	Row/Cell Hex Code	C G -Hanzi- T		J Kanji	K Hanja	
(1)......	078/000	*[graphic symbols are shown in this row]*				
(2)......	4E00	0-523B	1-4421	0-306C	0-6C69	.....(3)
		0-5027	1-3601	0-1676	0-7673	.....(4)

Key to example entry above:

(1) Row/Cell in decimal
(2) Code position in hexadecimal
(3) Source code - code position in hexadecimal
(4) Source code - section and position number

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

Each of the other columns shows the graphic symbol for the character, and its coded representation, as specified in a source standard for coded character sets that is also identified in the table entry. Each of these source standards is assigned to one of four groups indicated by G, T, J, or K 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, or K columns includes a sample graphic symbol from the source character set standard, together with its coded representation in that standard. The first line below the graphic symbol shows the coded representation in hexadecimal notation. The second line shows the coded representation in decimal notation which comprises two digits for section number followed by two digits for position number. Each of the coded representations is prefixed by a one-digit source code number followed by a hyphen. This source code number identifies the coded character set standard from which the character is taken as shown in the lists below.

Hanzi G sources are

G0	GB2312-~~19~~80
G1	GB12345-~~19~~90 with 58 Hong Kong and 92 Korean "Idu" characters
G3	GB7589-~~19~~87 unsimplified forms
G5	GB7590-~~19~~87 unsimplified forms
G7	General Purpose Hanzi List for Modern Chinese Language
G8	GB8565-~~19~~89

Hanzi T sources are

T1	TCA-CNS 11643/1st plane with some additional characters
T2	TCA-CNS 11643/2nd plane
TE	TCA-CNS 11643/14th plane with some additional characters

Kanji J sources are

J0	JIS X 0208-1990
J1	JIS X 0212-1990

Hanja K sources are

K0	KS C 5601-1987
K1	KS C 5657-1991

For CJK (Chinese/Japanese/Korean) ideographs in the BMP, the names shall be algorithmically constructed by appending their two-octet coded representation in hexadecimal notation to "CJK UNIFIED IDEOGRAPH-". For example, the first CJK ideograph character in the BMP has the name "CJK UNIFIED IDEOGRAPH-4E00".

N1396 A2

Annex A
(normative)

Collections of graphic characters for subsets

The following collections are from the Basic Multilingual Plane.

NOTE - Use of implementation levels 1 and 2 restricts the repertoire of some character collections (see 23.4). Collections which include combining characters are 7, 10, 13 to 26, 35, 49 , 50, 63 and 65.

Collection number and name	Positions
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 - 0652~~
14 BASIC ARABIC	0600 - 065F
~~15 ARABIC EXTENDED~~	~~0653 - 06FF~~
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
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	HANGUL	3400 - 3D2D
58	HANGUL SUPPLEMENTARY-A	3D2E - 44B7
59	HANGUL SUPPLEMENTARY-B	44B8 - 4DFF
60	CJK UNIFIED IDEOGRAPHS	4E00 - 9FFF
61	PRIVATE USE AREA	E000 - F8FF
62	CJK COMPATIBILITY IDEOGRAPHS	F900 - FAFF
63	ALPHABETIC PRESENTATION FORMS	FB00 - FB4F
64	ARABIC PRESENTATION FORMS-A	FB50 - 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

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

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

250	GENERAL FORMAT CHARACTERS	Collections 200 - 203
251	SCRIPT-SPECIFIC FORMAT CHARACTERS	Collections 204 - 206

The following specify other collections.

270	COMBINING CHARACTERS	characters specified in annex B.1
271	COMBINING CHARACTERS B-2	characters specified in annex B.2
300	BMP	0000 - D7FF E000 - FFFD

400	PRIVATE USE PLANES	G=00,
		P=0F, 10 & E0 - FF

500	PRIVATE USE GROUPS	G=60 - 7F

Annex B
(normative)

List of combining characters

B.1 List of all combining characters

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

0483	COMBINING CYRILLIC TITLO
0484	COMBINING CYRILLIC PALATALIZATION
0485	COMBINING CYRILLIC DASIA PNEUMATA
0486	COMBINING CYRILLIC PSILI PNEUMATA
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
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
064B	ARABIC FATHATAN
064C	ARABIC DAMMATAN
064D	ARABIC KASRATAN
064E	ARABIC FATHAH
064F	ARABIC DAMMAH
0650	ARABIC KASRAH
0651	ARABIC SHADDAH
0652	ARABIC SUKUN
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
06DD	ARABIC END OF AYAH
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
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
0A02	GURMUKHI SIGN BINDI
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
0A70	GURMUKHI TIPPI
0A71	GURMUKHI ADDAK
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
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
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
0B82	TAMIL SIGN ANUSVARA
0B83	TAMIL SIGN VISARGA
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
0C82	KANNADA SIGN ANUSVARA
0C83	KANNADA SIGN VISARGA
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
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
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
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
302A	IDEOGRAPHIC LEVEL TONE MARK
302B	IDEOGRAPHIC RISING TONE MARK
302C	IDEOGRAPHIC DEPARTING TONE MARK
302D	IDEOGRAPHIC ENTERING TONE MARK
302E	HANGUL SINGLE DOT TONE MARK
302F	HANGUL DOUBLE DOT TONE MARK
3099	COMBINING KATAKANA-HIRAGANA VOICED SOUND MARK
309A	COMBINING KATAKANA-HIRAGANA SEMI-VOICED SOUND MARK
FB1E	HEBREW POINT JUDEO-SPANISH VARIKA

B.2 List of characters not allowed in implementation level 2

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

NOTE - This list is a subset of the list in clause B.1 except for HANGUL JAMO (see clause 24).

0483	COMBINING CYRILLIC TITLO
0484	COMBINING CYRILLIC PALATALIZATION
0485	COMBINING CYRILLIC DASIA PNEUMATA
0486	COMBINING CYRILLIC PSILI PNEUMATA
093C	DEVANAGARI SIGN NUKTA
0953	DEVANAGARI GRAVE ACCENT
0954	DEVANAGARI ACUTE ACCENT
09BC	BENGALI SIGN NUKTA
09D7	BENGALI AU LENGTH MARK
0A3C	GURMUKHI SIGN NUKTA
0A70	GURMUKHI TIPPI
0A71	GURMUKHI ADDAK
0ABC	GUJARTI SIGN NUKTA
0B3C	ORIYA SIGN NUKTA
0B56	ORIYA AI LENGTH MARK
0B57	ORIYA AU LENGTH MARK
0BD7	TAMIL AU LENGTH MARK
0C55	TELUGU LENGTH MARK
0C56	TELUGU AI LENGTH MARK
0CD5	KANNADA LENGTH MARK
0CD6	KANNADA AI LENGTH MARK
0D57	MALAYALAM AU LENGTH MARK
302A	IDEOGRAPHIC LEVEL TONE MARK
302B	IDEOGRAPHIC RISING TONE MARK
302C	IDEOGRAPHIC DEPARTING TONE MARK
302D	IDEOGRAPHIC ENTERING TONE MARK
302E	HANGUL SINGLE DOT TONE MARK
302F	HANGUL DOUBLE DOT TONE MARK
3099	COMBINING KATAKANA-HIRAGANA VOICED SOUND MARK
309A	COMBINING KATAKANA-HIRAGANA SEMI-VOICED SOUND MARK

Annex C
(informative)

Mirrored characters in Arabic bi-directional context

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

0028	LEFT PARENTHESIS
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
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
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 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	REVERSE 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
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
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

Annex D
(informative)

Alternate format characters

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

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

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

D.1 General format characters

D.1.1 Zero-width boundary indicators

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

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

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

NOTE - For additional usages of this character for "signature", see annex F.

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

ZERO WIDTH NON-JOINER (200C): This character indicates that the adjacent characters ~~should~~ are not ~~be~~ 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 ~~should~~ are not ~~be~~ rendered with the normal cursive connection.

ZERO WIDTH JOINER (200D): This character indicates that the adjacent characters ~~should be~~ 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.

D.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 ~~should~~ starts; although the text ~~should~~ 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 ~~should~~ starts; e.g. the text ~~should~~ continues on the next line and inter-paragraph line spacing or paragraph indentation might be applied.

D.1.3 Bi-directional text formatting

The following characters are used in formatting bi-directional text. If the specification of a subset includes these characters, then text containing right-to-left characters are to be rendered with an implicit bi-directional 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 (Bi-directional 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 bi-directional formatting, this character acts like a left-to-right character (such as LATIN SMALL LETTER A). RIGHT-TO-LEFT MARK (200F): In bi-directional 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 (Bi-directional 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.

D.2 Script-specific format characters

D.2.1 Hangul fill characters

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

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

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

D.2.2 Symmetric swapping format characters

The following characters are used in conjunction with the class of left/right handed pairs of characters listed in clause 20. The following format characters indicate whether the interpretation of the term LEFT or RIGHT in the character names ~~should be~~ is OPENING or CLOSING respectively. The following characters do not nest.

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

INHIBIT SYMMETRIC SWAPPING (206A): Between this character and the following ACTIVATE SYMMETRIC SWAPPING format character (if any), the stored characters listed in clause 20 are ~~will be~~ interpreted and rendered as LEFT and RIGHT, and the processing specified in that clause ~~are~~ is not ~~to be~~ performed.

ACTIVATE SYMMETRIC SWAPPING (206B): Between this character and the following INHIBIT SYMMETRIC SWAPPING format character (if any), the stored characters listed in clause 20 are interpreted and rendered as OPENING and CLOSING characters as specified in that clause.

D.2.3 Character shaping selectors

The following characters are used in conjunction with Arabic presentation forms. During the presentation process, certain characters may be joined together in cursive connection or ligatures. The following characters indicate that the character shape determination process used to achieve this presentation effect is ~~to be~~ 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 ~~to be~~ inhibited. The stored Arabic presentation forms ~~will be~~ 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 ~~should be~~ 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.

D.2.4 Numeric shape selectors

The following characters allow the selection of the shapes in which the digits from 0030 to 0039 are ~~to be~~ 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 ~~will be~~ are rendered with the shapes as those shown in the code tables for those digits. This is the default state.

Annex F
(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 beginningof 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 recognises 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 recognise the characters correctly.

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

Annex G

[This Annex has been deleted, and its contents have been registered in the “ISO International Register of coded character sets to be used with escape sequences” as Registration no. 178 (revised).]

Annex H
(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 minimised between the receipt of a CC-data-element and its retransmission.

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

1.Receiving device with full retransmission capability

The originating device component will retransmit the coded representations of any received characters, including those that are outside the identified subset of the receiving device component, without change to their coded representation, unless modified by the user.

2.Receiving device with subset retransmission capability

The originating device component can retransmit only the coded representations of the characters of the subset adopted by the receiving device component.

Annex J
(informative)

Notations of octet value representations

Representation of octet values in ISO/IEC 10646 except in clause 17 is different from other character coding standards such as ISO/IEC 2022, ISO/IEC 6429 and ISO 8859. This annex clarifies the relationship between the two notations.

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

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

- In other character coding standards, the notation used to express an octet value is x/y, where x and y are two 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 K
(informative)

Character naming guidelines

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

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

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

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

Rule 1

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

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

Rule 2

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

Examples:
Name: LOCKING-SHIFT TWO RIGHT
Acronym: LS2R
Name: SOFT-HYPHEN
Acronym: SHY

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

Rule 3

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

The name of a character may also be followed by a single * symbol. This indicates that additional information on the character appears in Annex P. Any * symbols are omitted from the character names listed in Annex E.

Rule 4

The names of a character ~~shall~~ wherever possible denote its customary meaning, for examples PLUS SIGN. Where this is not possible, names ~~should~~ describe shapes, not usage; for example: UPWARDS ARROW.

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

Rule 5

Only one name is ~~will be~~ given to each character.

Rule 6

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

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

NOTES

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

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

Rule 7

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

Examples:

K LATIN CAPITAL LETTER K

IO CYRILLIC CAPITAL LETTER YU

[Ed.: replace IO *with Cyrillic* IO *]*

Rule 8

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

Examples:

# CYRILLIC CAPITAL LETTER I

I CYRILLIC CAPITAL LETTER BYELORUSSIAN-UKRAINIAN I

[Ed.: replace # with Cyrillic H*]*

Rule 9

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

Examples:

A LATIN CAPITAL LETTER A
A GREEK CAPITAL LETTER ALPHA
A CYRILLIC CAPITAL LETTER A

Rule 10

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

Example:

µ MICRO SIGN

Rule 11

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

Examples:

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

Rule 12

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

Example:

UNDERTIE (Enotikon)

Rule 13

The above rules ~~shall~~ do not apply to ideographic characters. These characters ~~will be~~ are identified by alpha-numeric identifiers specified for each ideographic character (see clause 26).

Annex L
(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 this part of 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 —Cyrillic alphabet coded character set for historic Slavonic languages and European non-Slavonic languages written in a Cyrillic script for bibliographic information interchange.

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

ISO 6937:1993, Information technology — Coded graphic character sets for text communication — Latin alphabet.

ISO 8859, Information processing 8-bit single-byte coded graphic character sets
-Part 1. Latin alphabet No. 1 (1987).
-Part 2. Latin alphabet No. 2 (1987).
-Part 3. Latin alphabet No. 3 (1988).
-Part 4. Latin alphabet No. 4 (1988).
-Part 5. Latin/Cyrillic alphabet (1988)
-Part 6. Latin/Arabic alphabet (1987)
-Part 7. Latin/Greek alphabet (1987)
-Part 8. Latin/Hebrew alphabet (1988)
-Part 9. Latin alphabet No. 5 (1989)
-Part 10. Latin alphabet No. 6 (1993).

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

ISO 8957:1993, 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 10367:1991, Information technology Standardized coded graphic character sets for use in 8-bit codes.

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 for information interchange.

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 Methodology. Data processing — 7-bit coded character set for information interchange.

GB2312-~~19~~80 Code of Chinese Graphic Character Set for Information Interchange: Jishu Biaozhun Chubanshe (Technical Standards Publishing).

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

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).

KS C 5601-1987 Korean Industrial Standards Association. Jeongho gyohwanyong buho (Hangul mit Hanja) (Code for Information Interchange (Hangul and Hanja)).

KS C 5657-1991 Korean Industrial Standards Association. Jeongho gyohwanyong buho hwakjang saten (Code of the supplementary Korean graphic character set for information interchange).

TIS 620-2533:1990 Thai Industrial Standard for Thai Character Code for Computer.

Esling, John. Computer coding of the IPA: supplementary report. Journal of the International Phonetic Association, 20:1 (1990), p. 22-26.

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 Phonetic Association, 19:2 (1989), p. 81-82.

International Phonetic Association. The International Phonetic Alphabet (revised to 1989).

Knuth, Donald E. The TeXbook. — 19th. printing, rev.— Reading, MA : Addison-Wesley, 1990.

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.

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.

Annex M
(informative)

External references to character repertoires

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

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

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

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

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

M.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 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 this part of ISO/IEC 10646 are identified by means of numbers (arcs) which follow the arcs "10646" and "1" which identify the part one of ISO/IEC 10646.

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

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

The second such arc identifies the repertoire subset, and is either:

- all (0), or
- collections (1).

Arc (0) identifies the entire collection of characters specified in this part of ISO/IEC 10646. No further arc follow this arc.

NOTE - 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 - As an example, the object identifier for the subset comprising the collections BASIC LATIN, LATIN-1 SUPPLEMENT, and MATHEMATICAL OPERATORS, at implementation level 1, is:

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

ISO/IEC 8824 also specifies object descriptors corresponding to object identifier values. For each combination of arcs the corresponding object descriptor are as follows:

1 0 : "ISO 10646 part-1 level-1 unrestricted"
2 0 : "ISO 10646 part-1 level-2 unrestricted"
3 0 : "ISO 10646 part-1 level-3 unrestricted"

For a single collection with collection name "xxx".

1 1 : "ISO 10646 part-1 level-1 xxx"
2 1 : "ISO 10646 part-1 level-2 xxx"
3 1 : "ISO 10646 part-1 level-3 xxx"

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

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

2 1 : "ISO 10646 part-1 level-2 collections m1,m2, m3, ... "

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

NOTE - All spaces are single spaces.

M.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 annex B, the coded representation form specified in this part of ISO/IEC 10646 is identified by means of numbers (arcs) which follow the arcs "10646" and "1" which identify this part of ISO/IEC 10646.

The first such arc is:
- transfer-syntaxes (0).

The second such arc identifies the form and is either:
- two-octet-BMP-form (2), or
- four-octet-form (4), or
- transformation-format-16 (5), or
- UTF8-form (8).

NOTE - As an example, the object identifier for the two-octet coded representation form is:

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

The corresponding object descriptors are:
- "ISO 10646 part-1 form 2" ~~and~~
- "ISO 10646 part-1 form 4"
- “ISO 10646 part-1 utf-16”
- “ISO 10646 part-1 utf-8”.

Annex N
(informative)

Scripts under consideration for future editions of ISO/IEC 10646

In order to make sure that ISO/IEC 10646 is useful for people using their native scripts, characters included in ISO/IEC 10646 were selected with input and feedback from national standards organisations and/or qualified experts.

Some scripts and symbols were not included in this edition because sufficient input and feedback have not been provided during the preparation and review stages. It is intended that character code positions for these scripts and symbols will be allocated when sufficient input and review is provided. Such scripts and symbols include:

- Burmese
- Cree and Inuktitut
- Ethiopian
- Extensions to various scripts for Indo-European languages
- Hieroglyphics
- Khmer
- Maldivian
- Mongolian
- Runic
- Sinhalese
- Syriac
- Tibetan
- Yi

This list is not exhaustive. Other scripts and symbols as well as additional characters for the included scripts are expected to be included in future editions of ISO/IEC 10646.

Annex P

(Informative)

Additional information on characters

Note: New entries are marked with % and are not underlined.

This Annex contains additional information on some of the characters specified in clauses 25 and 26 of this International Standard. This information is intended to clarify some feature of a character, such as its naming or usage, or its associated graphic symbol.

Each entry in this Annex consists of the name of a character and its code position in the two-octet form, followed by the related additional information. Entries are arranged in ascending sequence of code position.

When an entry for a character is included in this Annex an * symbol appears immediately following its name in the corresponding table in clause 25 or 26 of this International Standard.

Group 00, Plane 00 (BMP)

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 20. The graphic symbol associated with it may differ from that in Table 2. %

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 20. The graphic symbol associated with it may differ from that in Table 2. %

00C6 LATIN CAPITAL LETTER AE (ash)

In the first edition of this International Standard the name of this character was:

LATIN CAPITAL LIGATURE AE

00E6 LATIN SMALL LETTER AE (ash)

In the first edition of this International Standard the name of this character was:

LATIN SMALL LIGATURE AE

0189 LATIN CAPITAL LETTER AFRICAN D

This character is the capital letter form of:

0256 LATIN SMALL LETTER D WITH TAIL

019F LATIN CAPITAL LETTER O WITH MIDDLE TILDE

This character is the capital letter form of:

0275 LATIN SMALL LETTER BARRED O

01E2 LATIN CAPITAL LETTER AE WITH MACRON (ash)

In the first edition of this International Standard the name of this character was:

LATIN CAPITAL LIGATURE AE WITH MACRON

01E3 LATIN SMALL LETTER AE WITH MACRON (ash)

In the first edition of this International Standard the name of this character was:

LATIN SMALL LIGATURE AE WITH MACRON

01FC LATIN CAPITAL LETTER AE WITH ACUTE (ash)

In the first edition of this International Standard the name of this character was:

LATIN CAPITAL LIGATURE AE WITH ACUTE

01FD LATIN SMALL LETTER AE WITH ACUTE (ash)

In the first edition of this International Standard the name of this character was:

LATIN SMALL LIGATURE AE WITH ACUTE

06AF ARABIC LETTER GAF

The symbol for a Hamza (see position 0633) may appear in the centre of the graphic symbol associated with this character. %

06D0 ARABIC LETTER E

This character may be used as an Arabic letter Sindhi bbeh. %

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. %

FFE3 FULLWIDTH MACRON

This character is the full-width form of the character: 00AF MACRON. It may also be used as the full-width form of the character:

203E OVERLINE. %