International Standard ISO 2022

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION•МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ•ORGANISATION INTERNATIONALE DE NORMALISATION

Information processing — ISO 7-bit and 8-bit coded character sets — Code extension techniques

Traitement de l'information — Jeux ISO de caractères codés à 7 et à 8 éléments — Techniques d'extension de code

Third edition — 1986-05-01

ISO 2022-1986 (E)

Price based on 25 pages

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work.

Draft International Standards adopted by the technical committees are circulated to the member bodies for approval before their acceptance as International Standards by the ISO Council. They are approved in accordance with ISO procedures requiring at least 75 % approval by the member bodies voting.

International Standard ISO 2022 was prepared by Technical Committee ISO/TC 97, *Information processing systems*.

ISO 2022 was first published in 1973. This third edition cancels and replaces the second edition, of which it constitutes a technical revision.

Users should note that all International Standards undergo revision from time to time and that any reference made herein to any other International Standard implies its latest edition, unless otherwise stated.

Contents

Information processing — ISO 7-bit and 8-bit coded character sets — Code extension techniques

1 Scope

This International Standard specifies methods of extending the 7-bit code, remaining in a 7-bit environment or increasing to an 8-bit environment. These techniques are described in four inter-related clauses dealing respectively with

— the extension of the 7-bit code remaining in a 7-bit environment;

— the structure of a family of 8-bit codes;

— the extension of an 8-bit code remaining in an 8-bit environment;

— the relationship between the 7-bit code and an 8-bit code.

This International Standard also describes the structure of families of codes which are related to the code of ISO 646 by their structure.

Code extension techniques are classified and some classes are given a structure in this International Standard. Specific assignments of bit combinations to relate individual character sets and control functions with their invocation or designation are to be made in accordance with ISO 2375 (see annex A).

2 Field of application

While the 7-bit code of ISO 646 is the agreed code for information interchange, an 8-bit code as described in this International Standard is provided for information interchange within an 8-bit environment.

The 7-bit code of ISO 646 allows the representation of up to 128 characters. Additionally, ISO 646 allows the representation of other graphic characters by the combination of two or more graphic characters with the control characters BACKSPACE or CARRIAGE RETURN. In some instances, the character set of ISO 646 lacks sufficient control functions or graphic characters to satisfy the needs of an application. These needs may be satisfied by means of code extension which is the subject of this International Standard.

The principles established in this International Standard may be utilized to form supplementary code extension facilities. For example ISO 6429 has followed such a procedure to formulate some parameterized control functions.

This International Standard presents a review of the salient structure of the 7-bit code and then builds upon that structure to specify various means of extending the control function and

graphic sets of the code. It also specifies structures and techniques to construct and formalize codes related to the 7-bit code. These related codes are structured so as to allow application-dependent usage without preventing the interchangeability of data employing them. It describes

a) the structure of the 7-bit code;

b) the extension of the 7-bit code, remaining in a 7-bit environment and making use of code extension techniques;

c) the structure of a family of 8-bit codes, remaining compatible with the 7-bit structure;

d) the extension of an 8-bit code, remaining in a 8-bit environment, and making use of code extension techniques.

In order to use identical techniques in each of the above cases, and to facilitate conversion between them, standard rules for code extension are necessary. This has the advantage of

a) reducing the risk of conflict between systems required to inter-operate;

b) permitting provision for code extension in the design of systems;

c) providing standardized methods of calling into use agreed sets of characters;

d) allowing the interchange of data between 7-bit and 8-bit environments, etc.

Code extension techniques are designed to be used for data to be processed serially in a forward direction. Use of these techniques in strings of data which are processed other than serially in a forward direction or included in data formatted for fixed-length record processing may have undesirable results or may require additional special treatment to ensure correct interpretation.

3 Conformance

Full conformance to a standard means that all its requirements are met. For such conformance to be unique the standard must contain no options. This is typically the case for hardware standards.

This International Standard is of a different nature and as a result, it is only practicable to envisage limited conformance to it, as defined hereunder.

This International Standard addresses whole classes of provisions, and it is not intended that they are all implemented in all instances.

1

Under limited conformance, the following is required :

a) when the code extension techniques described in this International Standard are used, they shall be implemented by the control functions defined in this International Standard with the meaning and coded representation specified in this International Standard;

b) when two systems with different levels of implementation of code extension techniques are required to communicate with one another, they shall do so using the code extension techniques they have in common;

c) coded representation that is either reserved for registration and not assigned or reserved for future standardization shall not be used;

d) no registered escape sequence shall be used with a meaning different from that defined by the registration.

4   References

ISO 646, *Information processing — ISO 7-bit coded character set for information interchange.*

ISO 2375, *Data processing — Procedure for registration of escape sequences.*

ISO 4873, *Information processing — ISO 8-bit code for information interchange — Structure and rules for implementation.*

ISO 6429, *Information processing — ISO 7-bit and 8-bit coded character sets — Additional control functions for character-imaging devices.*

5   Definitions and notation

5.1   Definitions

For the purpose of this International Standard, the following definitions apply.

5.1.1   bit combination : An ordered set of bits that represents a character or is used as a part of the representation of a character.

5.1.2   byte : A bit string that is operated upon as a unit and the size of which is independent of redundancy or framing techniques.

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

5.1.4   coded character set; code : A set of unambiguous rules that establishes a character set and the one-to-one relationship between the characters of the set and their bit combinations.

5.1.5   code extension : The techniques for the encoding of characters that are not included in the character set of a given code.

5.1.6   code table : A table showing the character allocated to each bit combination in a code.

5.1.7   control character : A control function the coded representation of which consists of a single bit combination.

5.1.8   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 bit combinations.

5.1.9   to designate : To identify a set of characters that are to be represented, in some cases immediately and in others on the occurrence of a further control function, in a prescribed manner.

5.1.10   environment : The characteristic that identifies the number of bits used to represent a character in a data processing or data communication system or in part of such a system.

5.1.11   escape sequence : A bit string that is used for control purposes in code extension procedures and that consists of two or more bit combinations. The first of these bit combinations represents the character ESCAPE.

5.1.12   Final character : The character the bit combination of which terminates an escape sequence.

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

5.1.14   graphic symbol : A visual representation of a graphic character or of a control function.

5.1.15   Intermediate character : A character the bit combination of which occurs between that of the ESCAPE character and that of the Final character in an escape sequence consisting of more than two bit combinations.

5.1.16   to invoke : To cause a designated set of characters to be represented by the prescribed bit combinations whenever those bit combinations occur, until an appropriate code extension function occurs.

5.1.17   position : That part of a code table identified by its column and row coordinates.

5.1.18   to represent :

a) to use a prescribed bit combination with the meaning of a character in a set of characters that has been designated and invoked; or

b) to use an escape sequence with the meaning of an additional control function.

5.1.19 version of the 7-bit code : A 7-bit coded character set in which all options left open in ISO 646 have been exercised. A single character shall be allocated to each of the bit combinations for which this freedom exists or the bit combination shall be declared unused.

5.1.20 version of the 8-bit code : An 8-bit coded character set in which all options left open in ISO 4873 have been exercised. A single character shall be allocated to each of the bit combinations for which this freedom exists or the bit combination shall be declared unused.

5.2 Notation

In this International Standard the following notations are used :

Bits of a 7-bit combination	—	b_7	b_6	b_5	b_4	b_3	b_2	b_1
Bits of an 8-bit combination	b_8	b_7	b_6	b_5	b_4	b_3	b_2	b_1
Bit weight for column and row reference	2^3	2^2	2^1	2^0	2^3	2^2	2^1	2^0
		Column			Row			

A bit combination is sometimes referred to by the column and row numbers of its position in the code table. The column number is the decimal equivalent of bits b_7 to b_5 (or b_8 to b_5) and the row number is the decimal equivalent of bits b_4 to b_1, giving to these bits the weights shown above. The column and row numbers are separated by a solidus.

In representing the decimal equivalents, the convention is to append a leading zero to the column number for 8-bit columns 00 to 09. As an example, the position of SPACE in the 7-bit code table is 2/0; the position of the same character in an 8-bit code table is 02/0.

6 Extension of the 7-bit code remaining in a 7-bit environment

6.1 Introduction

6.1.1 Structure of the 7-bit code

The 7-bit code table which is the basis of code extension techniques for use with the 7-bit coded character set of ISO 646 consists of areas for an ordered set of control characters and graphic characters grouped as follows :

   a) columns 0 and 1 contain a set of 32 control characters;

   b) columns 2 to 7 contain either :

     1) the character SPACE in position 2/0, which may be regarded as a control character or a graphic character, the character DELETE in position 7/15 and a set of 94 graphic characters in positions 2/1 to 7/14;

or

     2) a set of 96 graphic characters in positions 2/0 to 7/15.

This is shown in figure 1.
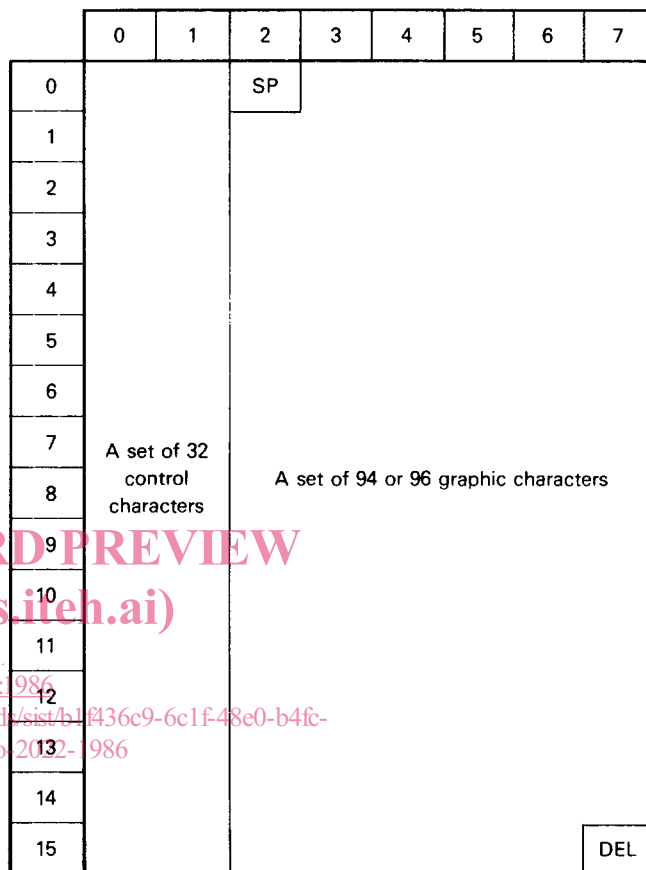
Figure 1 — 7-bit code structure

6.1.2 Extension by substitution

In many cases, the provisions of ISO 646 satisfy the requirements of an application. Other applications are satisfied by the use of a similarly structured code in which some of the characters of ISO 646 are substituted by other characters. Such substitution shall be regarded as constituting a new code, outside the provisions of ISO 646.

6.1.3 Extension by increasing the repertoire of characters

This International Standard provides for characters additional to the 128 provided by the structure of the 7-bit code in the following ways :

   a) additional single control functions;

   b) additional sets of 32 control functions;

   c) additional sets of 94 graphic characters;

d) additional sets of 96 graphic characters;

e) additional sets of more than 94 or 96 graphic characters, each represented by more than one byte, i.e. multiple-byte sets.

Any additional set of characters, described in c), d) and e) above, shall not contain SPACE or any control character, such as DELETE.

6.1.4 Elements of code extension

Many applications require combinations of the above code extension facilities. The elements of code extension are shown in figure 2, where the names of elements are defined as follows :

a) C0 set : a set of 32 control characters (columns 0 and 1);

b) C1 set : an additional set of 32 control functions;

c) other additional single control functions;

d) G0 set : a set of 94 graphic characters (bit combinations 2/1 to 7/14), a multiple-byte set may also function as a G0 set;

e) G1, G2, G3 sets : additional sets of 94 graphic characters (bit combinations 2/1 to 7/14) or of 96 graphic characters (bit combinations 2/0 to 7/15); a multiple-byte set may also function as a G1, G2 or G3 set.

NOTE — It is intended that, if they are used, a set of control characters and a set of graphic characters which are compatible with ISO 646 (see 6.1.5) are assigned to the C0 set and the G0 set respectively.

6.1.5 Compatibility

For the purpose of interchange, various levels of compatibility, which may be preserved when applying extension facilities, are identified. The following three such levels are distinguished in this International Standard :

a) a version according to ISO 646;

b) a compatible variant of ISO 646, i.e. a 7-bit code which is compatible with ISO 646 inasmuch as

— columns 0 and 1 contain only control characters;

— the ten transmission control characters and NUL, SO, SI, CAN, SUB, ESC, SP and DEL remain unaltered in their meanings and in their positions in the code table;

— SPACE, DELETE and one or two sets of 94 graphic characters are allocated to columns 2 to 7; sets of 96 graphic characters, multiple-byte sets and shift functions other than SO, SI are not used;

— graphic characters of ISO 646 are not moved to other positions (a non-Latin alphabet containing graphic characters which are also included in the Latin alphabet is not subject to this rule).

c) other 7-bit codes structured as in 6.1.1. Such a code may contain 94 graphic characters, 96 graphic characters and/or multiple-byte sets in columns 2 to 7. To be able to provide the facilities of code extension of this International Standard, the control characters ESCAPE, SHIFT-OUT and SHIFT-IN shall remain unaltered in their meanings and their positions in the code table.
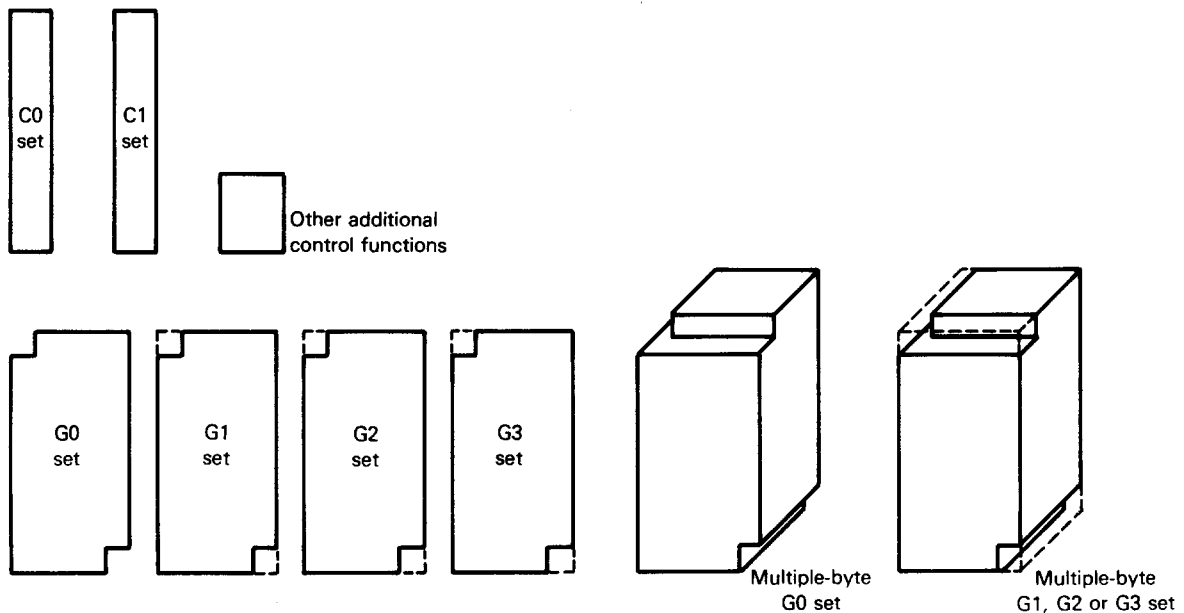

Figure 2 — Code extension elements

4

6.1.6 Code extension characters of ISO 646

In ISO 646 the following control characters are provided for the purpose of code extension :

— ESCAPE	ESC
— SHIFT-OUT	SO
— SHIFT-IN	SI
— DATA LINK ESCAPE	DLE

This International Standard does not describe the use of the control character DATA LINK ESCAPE which is reserved for the provision of additional transmission control functions. The use of this character is specified in other International Standards.

6.1.7 Other code extension characters

For use within a 7-bit environment, this International Standard includes provision of some additional shift functions which are not included in ISO 646 :

— LOCKING-SHIFT TWO	LS2
— LOCKING-SHIFT THREE	LS3
— SINGLE-SHIFT TWO	SS2
— SINGLE-SHIFT THREE	SS3

See annex B for the coded representation of these functions.

Three additional locking-shift functions LS1R, LS2R, LS3R are specified in 8.2.1; they are used in a 7-bit environment only when it is necessary to preserve their use for transformation between 7-bit and 8-bit environments (see 10.2 and 10.4). When used in a 7-bit code, LS1R, LS2R and LS3R have the same effects as SO, LS2 and LS3, respectively.

6.1.8 Combination of graphic characters

Some graphic sets may allow for the representation of additional graphic characters such as accented characters by the combination of two or more graphic characters in the same character position. Two methods of combining graphic characters in a single character position are provided for

a) graphic characters having implicit forward motion (spacing characters) used in conjunction with BACKSPACE or CARRIAGE RETURN;

b) graphic characters having no implicit forward motion (non-spacing characters) used in combination with spacing graphic characters.

ISO 646 allows for the first of these two methods to represent accented characters. Sponsors of graphic sets applying for registration under the provision of ISO 2375 are expected to identify any characters in the set that are non-spacing.

NOTE — A standard defining a character set should specify any restriction on combining characters, as this is not part of registration.

6.2 Extension of the graphic set by means of shift functions

For use in a 7-bit environment, the shift functions specified in this International Standard are

SO, SI, LS2, LS3, SS2 and SS3.

6.2.1 Use of locking-shift functions

In a 7-bit environment, the functions SHIFT-OUT (SO), SHIFT-IN (SI), LOCKING-SHIFT TWO (LS2) and LOCKING-SHIFT THREE (LS3) shall be used exclusively for extension of the graphic set.

The shift functions SO, LS2 or LS3 shall each invoke an additional set of 94 or 96 graphic characters : G1, G2 and G3. If the set consists of 94 characters, it is invoked into positions 2/1 to 7/14, SPACE is invoked into position 2/0 and DELETE is invoked into position 7/15; if the set consists of 96 characters, it is invoked into positions 2/0 to 7/15. Graphic characters need not be assigned to all the positions of the additional set, nor, except as specified below, need all the graphic characters of the additional set be different from the graphic characters of the previously invoked set.

The shift function SI shall invoke the 94 graphic characters of the G0 set and cause positions 2/0 and 7/15 to take their normal meanings of SPACE and DELETE, respectively.

If a particular set is already invoked, use of the corresponding shift function has no effect.

The meanings of the following bit combinations shall not be affected by the occurrence of the locking-shift functions :

a) those representing the control characters in columns 0 and 1;

b) those included in any escape sequence;

c) the one following SS2 or SS3.

The characters SPACE and DELETE shall occur at positions 2/0 and 7/15, respectively, if and only if a set of 94 graphic characters is invoked. They shall not be assigned to any other positions in any set. However, characters other than SPACE and representing spaces of different sizes or usage may be assigned to any positions in any set of graphic characters or control functions.

At the beginning of any information interchange the shift status shall be defined by use of one of the locking-shift functions as specified in 6.4 (see also clause 9).

6.2.2 Use of single-shift functions

The single-shift functions SS2 and SS3 shall be used exclusively for extension of the graphic set. SS2 shall invoke one character from the last designated G2 set. SS3 shall invoke one character from the last designated G3 set.

These invocations alter the meaning of the immediately following bit combination only (but see 6.3.9) and ascribe to it the meaning of the corresponding bit combination of the G2 or G3 set. The bit combination permitted to follow SS2 or SS3 is one

of those from 2/1 to 7/14 for a 94-character G2 or G3 set and 2/0 to 7/15 for a 96-character G2 or G3 set (see 10.3). The use of a single-shift function does not affect the current shift status established by a locking-shift function.

6.2.3 Unique additional graphic sets

Some applications require no more than three additional graphic sets of 94 or 96 characters that can be uniquely identified as G1, G2 and G3 sets. These sets are designated by means of appropriate escape sequences as described in 6.3.7 to 6.3.10. As described in 6.4, such sequences may be omitted by agreement between interchanging parties. Any of these additional sets can then be invoked by means of the corresponding shift functions.

6.2.4 Multiple graphic sets

If there is a requirement for more than three additional graphic sets or for more than one graphic set to be designated as either G0, G1, G2 or G3, it is necessary to designate the G0, G1, G2, G3 sets to be used next by means of the appropriate escape sequences as described in 6.3.7 to 6.3.10. Each subsequent use of a shift function shall invoke the corresponding currently designated set.

It is not necessary to revert to the G0 set by use of SI before designating a different set as G1, G2, G3 by means of an escape sequence.

The use of a shift function shall invoke the graphic characters of the set last designated for use by that shift function but shall not affect the identity of any sets currently designated. A designated set may be invoked any number of times by repeated use of the relevant shift function until it is superseded by another designating escape sequence.

When a further graphic set is designated by an escape sequence, the current shift status shall remain unaltered.

When a graphic set is designated by an escape sequence, and if that class of graphic set (i.e. G0, G1, G2 or G3) is currently invoked, then the new set shall also be invoked.

Figure 3 is a schematic representation of the designation and invocation processes described above.

6.3 Code extension by means of escape sequences

6.3.1 Purposes of escape sequences

Escape sequences provide single or sets of control functions other than for transmission control. Escape sequences are also used to designate sets of graphic characters, different uses of some or all of the 7-bit code combinations, and coded character sets with a number of bits other than 7.

Thus escape sequences are required to provide, for example :

  a) a single control function not already in the code;

  b) a set of control functions not already in the code;

  c) a set of graphic characters not already in the code;

  d) a code structure different from that of the code.

6.3.2 Structure of escape sequences

An escape sequence shall consist of two or more 7-bit combinations. The first shall always be the bit combination representing ESCAPE and the last shall always be that representing the Final character. An escape sequence may also contain any number of 7-bit combinations representing Intermediate characters.

The meaning of an escape sequence shall be determined by the 7-bit combination representing its Intermediate character(s), if any, and by the 7-bit combination representing its Final character.

Intermediate characters are the 16 characters of column 2 of the 7-bit code table; they are denoted by the symbol I.

Final characters are the 79 characters of columns 3 to 7 of the 7-bit code table excluding position 7/15; they are denoted by the symbol F.

NOTE — Although, in this International Standard, escape sequences are described in terms of characters or of positions in the code table, the meaning of an escape sequence is determined only by its bit combinations and it is unaffected by any meaning assigned to these bit combinations taken individually.

The control characters in columns 0 and 1 and the character in position 7/15 shall not be used as either Intermediate or Final characters to construct an escape sequence.

NOTE — As these prohibited characters may appear in an escape sequence in error, it may be necessary within an application to provide methods of identifying such a situation and of recovering from it, but this is not covered by this International Standard.

6.3.3 Categories of escape sequences

The use of escape sequences is specified in this International Standard. However, escape sequences with Final characters from column 3 are reserved for private use subject to the categorization outlined below. Escape sequences for private use are not subject to registration under ISO 2375.

NOTE — The implementors of any private escape sequence described as such in this International Standard are alerted to the fact that other implementors may give different meanings to the same escape sequence or may use different escape sequences to mean the same thing. Furthermore, such meanings may subsequently be assigned to registered escape sequences. Interchanging parties are warned that the use of such private escape sequences may reduce their capability to interchange data subsequently.

6.3.3.1 Two-character escape sequences

A two-character escape sequence shall be of the form

  ESC F

Such escape sequences are used to represent additional control functions.