International Standard 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

UDC 681.3.042 **Ref. No. ISO 2022-1986 (E)**

Descriptors : data processing, data transmission, coded character sets, code extensions.

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.

Printed in Switzerland

Contents

Annexes

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.

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.

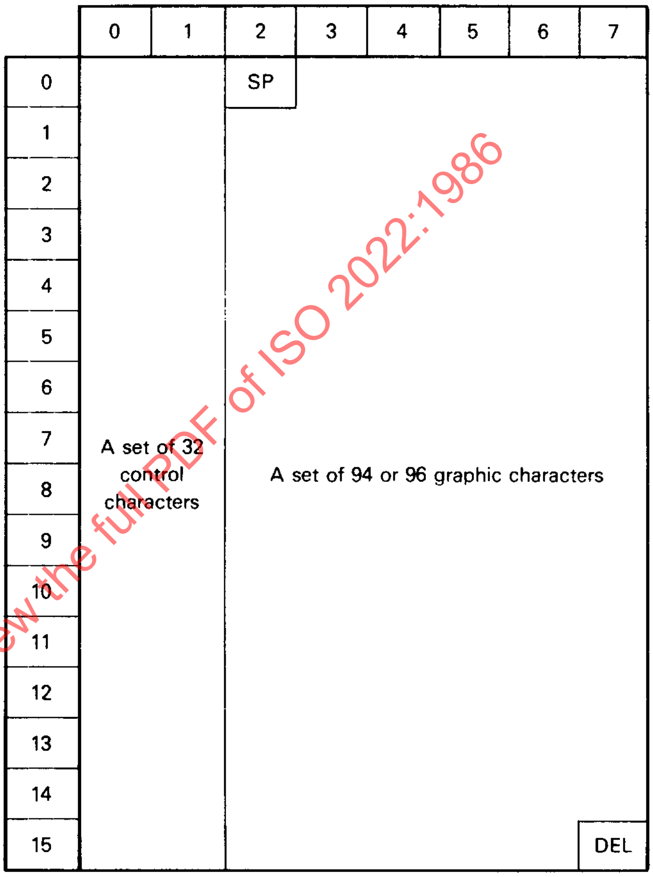

0
1
2
3
4
5
6
7
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
SP
A set of 32 control characters
A set of 94 or 96 graphic characters
DEL

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 twc 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.

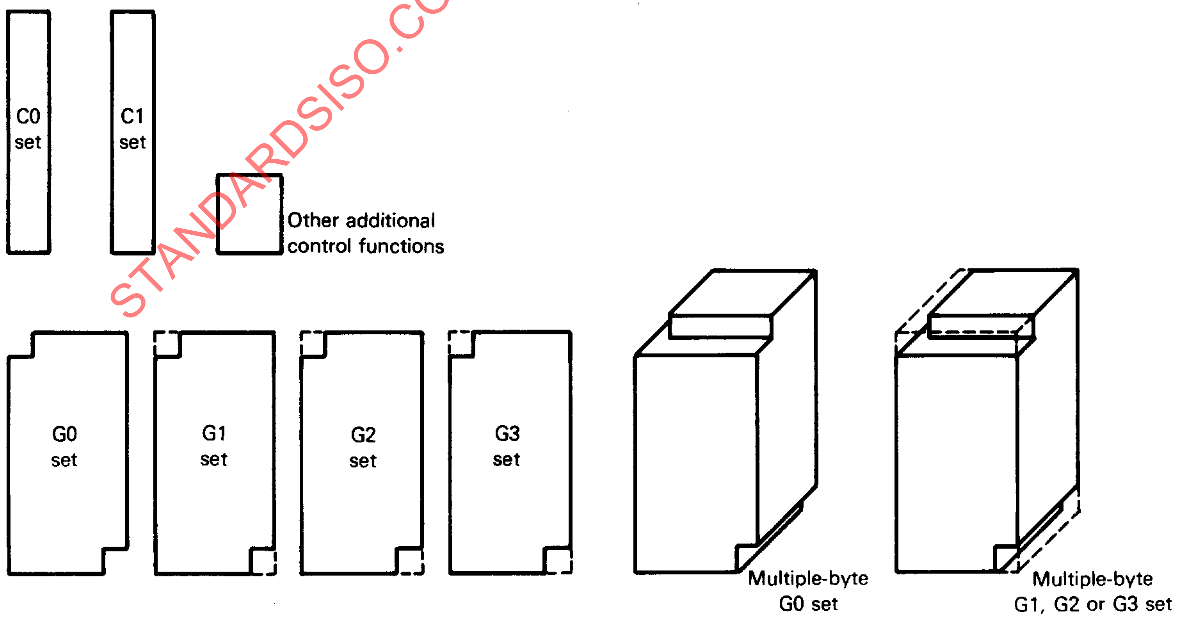


Figure 2 — Code extension elements

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.

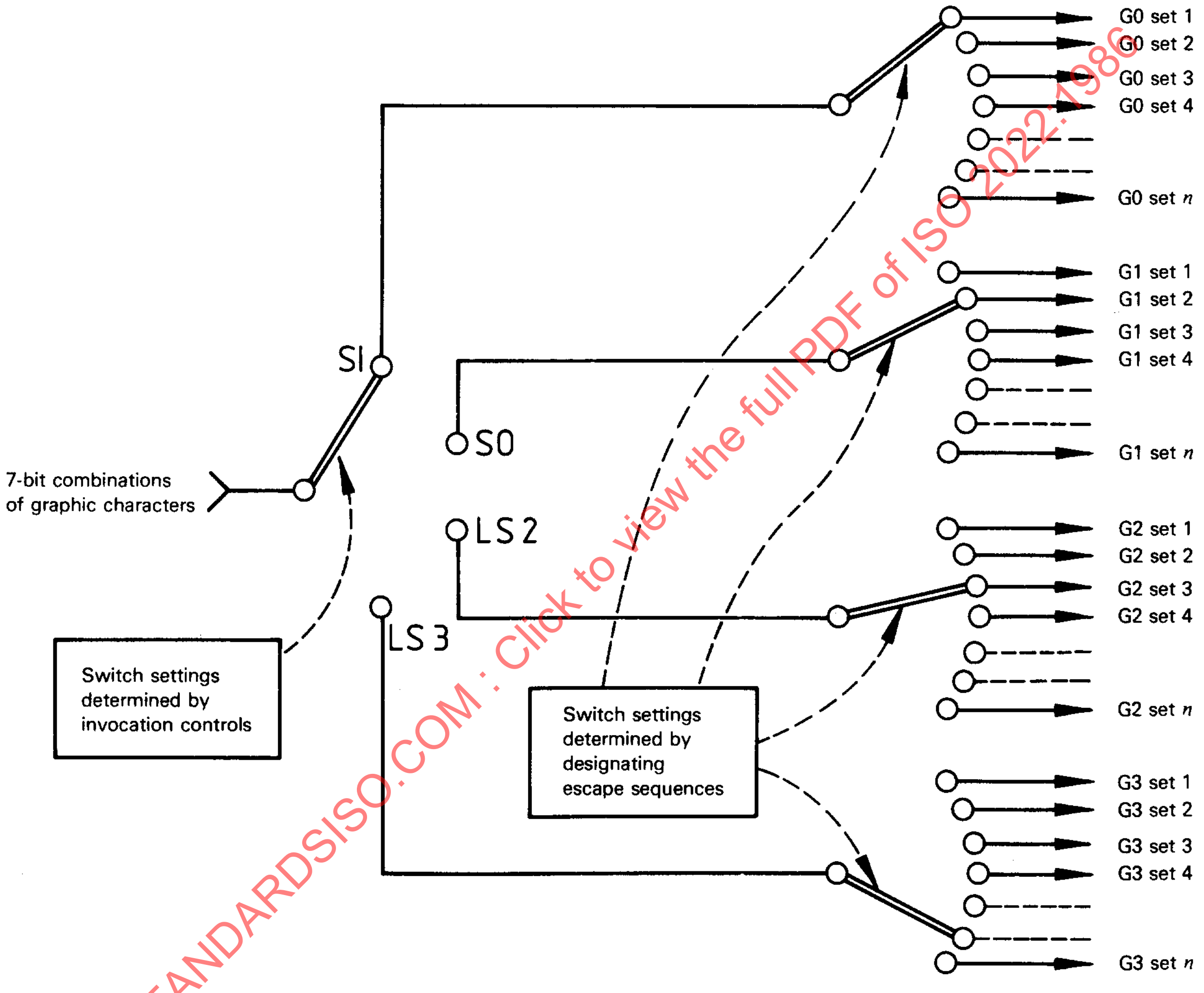


Figure 3 — Multiple graphic sets used with locking-shift functions

The 79 two-character sequences are split into three types, depending on the Final character, as shown in figure 4.

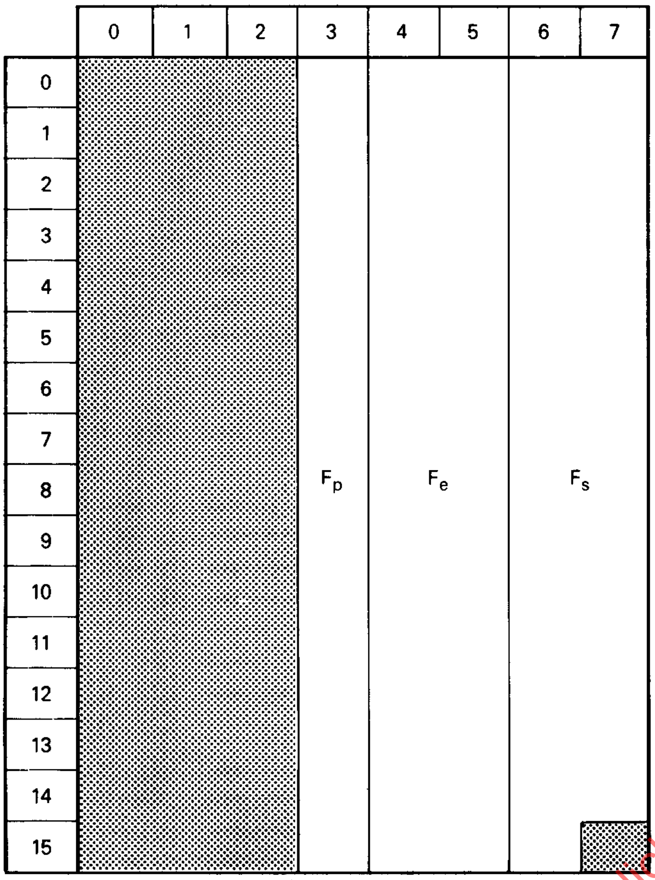


Figure 4 — Final character for two-character escape sequences

An ESC F_s sequence represents, depending on the Final character used, a single additional control function with a permanently assigned meaning. Thirty-one Final characters of columns 6 and 7 are provided for this purpose.

ESC F_s sequences are registered in the ISO International Register of Coded Character Sets to be used with escape sequences, which is maintained by the Registration Authority (see annex A). Any candidates for ESC F_s sequences shall be approved by ISO/TC 97/SC 2 for registration. The coding for the Final character, F_s, is assigned by the Registration Authority according to the procedure of ISO 2375.

An ESC F_e sequence represents, depending on the Final character used, an individual control function of the currently designated C1 set of 32 control functions (see 6.3.6). The 32 Final characters of columns 4 and 5 are provided for this purpose. Some applications require the use of only one such additional set. In this case, the set is identified either by the appropriate escape sequence, as described in 6.3.6, or by agreement between the interchanging parties. If more than one additional set of control functions is required to co-exist in a system, the set to be used next is designated and invoked by the appropriate escape sequence.

An ESC F_p sequence represents, depending on the Final character used, a single additional control function without standardized meaning for private use as required, subject to the prior agreement of the sender and the recipient of the data. The 16 Final characters of column 3 are provided for this purpose.

6.3.3.2 Three-character escape sequences

A three-character escape sequence shall be of the form

ESC I F

All types of three-character escape sequences are grouped into classes, according to their purpose, by means of their Intermediate characters, as shown in 6.3.4 to 6.3.13 (see table 1).

These sequences are split into two types according to their Final character as shown in figure 5.

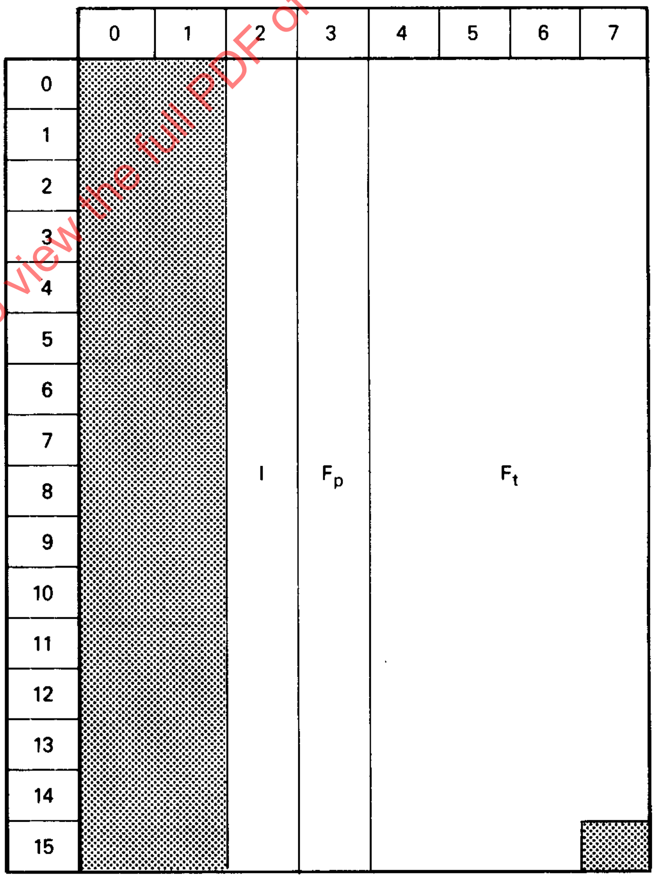


Figure 5 — Intermediate and Final characters for three-character escape sequences

ESC I F_t sequences are used for standardized purposes. Sixty-three F_t characters of columns 4 to 7 are provided for this purpose. One such Final character, that represented by bit combination 7/14 (when used with Intermediate characters 2/1, 2/2, 2/4, 2/8 to 2/11 or 2/13 to 2/15), shall indicate that the character set designated by the escape sequence is empty, i.e. that it does not contain any character. Bit combinations representing characters of a set which has been declared empty shall not be used. ESC I F_p sequences are reserved for private use. Sixteen F_p characters of column 3 are provided for this purpose.

6.3.3.3 Escape sequences having four or more characters

An escape sequence having four or more characters shall be of the form

ESC I ... I F

where I ... I represents two or more characters.

An escape sequence having four or more characters shall be interpreted as follows :

a) the first Intermediate character will indicate the same class of use as the Intermediate character of a three-character escape sequence;

b) except when the first Intermediate character is reserved for future standardization or is 2/0, 2/4, 2/5 or 2/6 (see 6.3.12, 6.3.9, 6.3.11 and 6.3.13 or clause 8, respectively), the second Intermediate character shall be used as follows :

— 2/0 is reserved for future standardization when the first Intermediate character is 2/1, 2/2 or 2/3 and is reserved for designation of DRCS when the first Intermediate character is one of 2/8 to 2/11 or 2/13 to 2/15 (see 6.3.10);

— 2/1 to 2/3 and any further Intermediate characters are available for registration of control functions and graphic character sets;

— 2/4 to 2/15 are reserved for future standardization.

c) all escape sequences that have a Final character of the F_p type are reserved for private use and are not specified by this International Standard;

d) the use of 7/14 as a Final character to signify an empty set, as specified in 6.3.3.2, also applies to escape sequences having four or more characters.

6.3.4 Single additional control functions

ESC 2/3 F represents a single additional control function determined by the Final character used.

6.3.5 Sets of 32 control characters for columns 0 and 1

ESC 2/1 F designates and invokes the C0 set of 32 control characters for representation by the bit combinations of columns 0 and 1.

The 10 transmission control characters, when included in a C0 set, shall retain their meanings and their positions in the code table. No other transmission control characters may be included in a C0 set.

To reduce the risk of conflict in the interchange of data, this set should have the following characteristics :

a) inclusion of the 10 transmission control characters;

b) inclusion of the control characters NUL, SO, SI, CAN, SUB and ESC with their meanings and their positions in the 7-bit code table unaltered.

NOTE — Consideration should be given to the effect that changing the meaning of control characters can have on equipment when interchanging data. For example, the bit combination corresponding to HT will have the effect of "horizontal tabulation" to a system designed to respond to this control character.

6.3.6 Sets of 32 control functions for representation by ESC F_e

ESC 2/2 F designates and invokes the C1 set of 32 control functions without affecting the C0 set.

Individual control functions of such a set are represented by means of ESC F_e sequences instead of a single bit-combination. A C1 set shall not include transmission control functions (see the note to 6.3.8).

6.3.7 Sets of 94 graphic characters

ESC 2/8 F designates a set of 94 graphic characters as the G0 set. The designated set is invoked by SI.

ESC 2/9 F designates a set of 94 graphic characters as the G1 set. The designated set is invoked by SO.

ESC 2/10 F designates a set of 94 graphic characters as the G2 set. LS2 invokes the designated set and SS2 invokes one character from the designated set.

ESC 2/11 F designates a set of 94 graphic characters as the G3 set. LS3 invokes the designated set and SS3 invokes one character from the designated set.

6.3.8 Sets of 96 graphic characters

ESC 2/13 F designates a set of 96 graphic characters as the G1 set. The designated set is invoked by SO.

ESC 2/14 F designates a set of 96 graphic characters as the G2 set. LS2 invokes the designated set and SS2 invokes one character from the designated set.

ESC 2/15 F designates a set of 96 graphic characters as the G3 set. LS3 invokes the designated set and SS3 invokes one character from the designated set.

NOTE — When sets of characters are registered, a unique Final character is allocated to each set. In the case of control character sets, the series of Final characters for C0 sets and C1 sets are quite separate — i.e. a set is registered for use as either a C0 or a C1 set. In contrast, graphic sets are not registered as either G0, G1, G2 or G3 sets but as all four. They may be used in any of these ways by use of the appropriate Intermediate character as specified in 6.3.7, 6.3.8 and 6.3.9, except that a set of 96 characters cannot be used as a G0 set.

6.3.9 Set of graphic characters with multiple-byte representation

ESC 2/4 I F designates a set of graphic characters that are represented by two or more bytes, each corresponding to a bit combination in columns 2 to 7 (see figure 6).

ESC 2/4 2/8 F designates a multiple-byte graphic set as the G0 set. The designated set is invoked by SI.

ESC 2/4 2/9 F or ESC 2/4 2/13 F designates a multiple-byte graphic set as the G1 set. The designated set is invoked by SO.

ESC 2/4 2/10 F or ESC 2/4 2/14 F designates a multiple-byte graphic set as the G2 set. LS2 invokes the designated set and SS2 invokes one character from the designated set.

ESC 2/4 2/11 F or ESC 2/4 2/15 F designates a multiple-byte graphic set as the G3 set. LS3 invokes the designated set and SS3 invokes one character from the designated set.

As an exception to these rules ESC 2/4 4/0, ESC 2/4 4/1 and ESC 2/4 4/2 designate multiple-byte sets as G0 sets, because they are already registered.

NOTE — The reason for these exceptions is that the first version of ISO 2022 allowed multiple-byte sets to be only G0 sets and used ESC 2/4 F to represent them.

A multiple-byte set designated by ESC 2/4 2/8 F, ESC 2/4 2/9 F, ESC 2/4 2/10 F or ESC 2/4 2/11 F consists of up to 94^n characters each of which is represented by a sequence of n bit combinations in the range 2/1 to 7/14. A multiple-byte set designated by ESC 2/4 2/13 F, ESC 2/4 2/14 F, or ESC 2/4 2/15 F consists of up to 96^n characters each of which is represented by a sequence of n bit combinations in the range 2/0 to 7/15. Within such a set, each graphic character is represented by the same number of bytes.

If a single-shift function is used to invoke a character from a multiple-byte set, contrary to its normal usage the shift function will affect two or more bit-combinations to represent that character.

Escape sequences with 2/4 as the first Intermediate character and either 2/0 to 2/7 or 2/12 as the second Intermediate character are reserved for future standardization.

The Final character assignment is as follows :

Final character from column	Number of bytes
3	2 or more for private use
4 and 5	2
6	3
7	4 or more

In an escape sequence designating a multiple-byte set, a third Intermediate character in the range 2/1 to 2/3 and any further Intermediate characters can be used after the second Intermediate character, if more than 63 sets need to be registered.

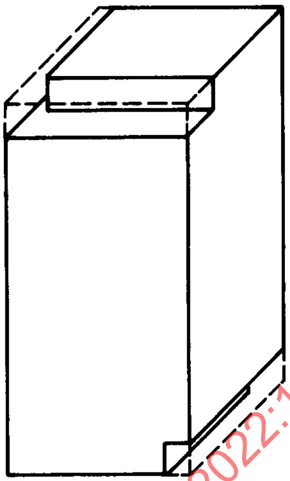


Figure 6 — Set of graphics with multiple-byte representation

6.3.10 Dynamically Redefinable Character Sets (DRCS)

A DRCS is a set of graphic characters the visual appearance of which is specified and transmitted prior to the time of use. Such a specification may be made explicitly or by a reference. These characters may be alphabetic, special symbols or picture element symbols. Once specified, a DRCS is regarded as a member of the repertoire of graphic character sets that can be designated by appropriate escape sequences as a G0, G1, G2 or G3 set.

ESC I 2/0 F designates such a set, where I shall be in the range 2/8 to 2/11 to indicate whether a 94-character set is to be used as a G0, G1, G2 or G3 set respectively or in the range 2/13 to 2/15, to indicate whether a 96-character set is to be used as a G1, G2 or G3 set respectively, in the same way as defined in 6.3.7 and 6.3.8.

One hundred and twenty-six, i.e. 2 × 63 sets may be identified by means of such four character sequences. This should be enough for most requirements but one or more additional Intermediate characters can be inserted between the second Intermediate character and the Final character if more sets are needed.

Escape sequences with 2/0 to 2/7 or 2/12 as the first and 2/0 as the second Intermediate character are reserved for future standardization.

Multiple-byte graphic sets may also be dynamically redefinable. ESC 2/4 I 2/0 F designates such a set, where I takes the same values (2/8 to 2/11 or 2/13 to 2/15) and has the same meanings as in the preceding paragraphs. Further Intermediate characters can be used if more than 63 sets need to be identified.

NOTES

1 This class of escape sequences is exceptional because the allocation of Final (and possibly Intermediate) characters is not done by the Registration Authority (see annex A) but by the user. It is recommended that Final characters be allocated sequentially, starting with 4/0.

2 The need for this particular escape sequence as distinct from the normal three-character sequence used to represent registered sets is that it implies exact description of the shape or font of the characters.

6.3.11 Other coding systems

ESC 2/5 F and ESC 2/5 I F (except ESC 2/5 4/0) designate and invoke a coding system different from that of this International Standard, not necessarily a character code.

The escape sequence ESC 2/5 4/0 is allocated to, and recommended for, use by such other coding systems for returning to the coding system of this International Standard. ESC 2/5 4/0 restores the state of the coding system to that at the time of invocation of the other coding system, that is announcers, designated and invoked control and graphic character sets. Whether or not other states, for example the active position, are restored is outside the scope of this International Standard.

In order to indicate whether the other coding system uses the return escape sequence ESC 2/5 4/0, the designating escape sequences are divided into the following categories :

> ESC 2/5 F shall mean that the other coding system uses ESC 2/5 4/0 to return;
>
> ESC 2/5 2/15 F shall mean that the other coding system does not use ESC 2/5 4/0 to return (it may have an alternative means to return or none at all).

In the first case, a second Intermediate character in the range 2/1 to 2/3 and any further additional Intermediate characters can be used, if more than 63 coding systems need to be registered.

In the second case, any further Intermediate characters can be used, if more than 63 coding systems need to be registered.

Escape sequences with 2/5 as the first Intermediate character and either 2/0 or 2/4 to 2/14 as the second Intermediate character are reserved for future standardization.

The above facility provides a means for switching between ISO 2022 coding systems and other coding systems when it is not performed at a higher level (see clause 10).

6.3.12 Announcement of extension facilities

ESC 2/0 F announces the extension facilities used in conjunction with data which follow. The use of these sequences is specified in clause 9.

6.3.13 Revision of registered sets

Annex A of this International Standard refers to the International Register of Coded Character Sets to be used with escape sequences established under the provisions of ISO 2375. ESC 2/6 F, when used, shall immediately precede a designating escape sequence and indicate a revision of a registered set. F will specify the revision number 1 to 63 by taking values 4/0 to 7/14 respectively. Revisions are only permitted to add a character or characters to a set and shall be submitted to the Registration Authority as required by ISO 2375, pointing out that the submission is a revision of a registered set. If the revision is not upwards compatible with the previous version, then an entirely new designating escape sequence shall be allocated.

NOTE — The combined use of the "revision number" escape sequence and of the original designating escape sequence facilitates the recognition by older devices or systems of newer versions of character sets.

Escape sequence with 2/6 as the first Intermediate character and any further Intermediate characters are reserved for future standardization.

6.3.14 Three-character escape sequences without assigned meanings

The escape sequences ESC 2/7 F and ESC 2/12 F have not been assigned meanings and are reserved for future standardization.

6.3.15 Summary of assignments of Intermediate characters

Table 1 summarizes the assignments of the Intermediate characters in the escape sequences. The shaded area denotes the combinations reserved for future standardization.

6.4 Initial designation and invocation

At the beginning of any information interchange, except where interchanging parties have agreed otherwise, all designations shall be defined by use of the appropriate escape sequences, and the shift status shall be defined by the use of the appropriate locking-shift functions. Interchanging parties who agree not to use such designators are warned that they may thereby reduce their capability to interchange data subsequently.

6.5 Pictorial representation of code extension in a 7-bit environment

Figure 7 summarizes, in a schematic form, the standard means of code extension available in a 7-bit environment.

7 Structure of a family of 8-bit codes

The family of 8-bit codes specified in this International Standard is obtained by the addition of one bit to each of the bit combinations of the 7-bit code, thus producing a set of 256 8-bit combinations. The characters of the 7-bit set are assigned to the 128 bit combinations the eighth bit of which is set to ZERO. In this way, the set as defined in 6.1 forms a defined and integral part of an 8-bit code that is structured in accordance with this International Standard. The 128 additional bit combinations, the eighth bit of which is set to ONE, are available for further assignment.

7.1 The 8-bit code table

A 16-by-16 array of columns numbered 00 to 15 and rows numbered 0 to 15 contains 256 code positions (see figure 8).

Columns 00 to 07 of this array contain 128 character positions which are in one-to-one correspondence with the characters of the 7-bit set. Their coded representation is the same as in the 7-bit environment with the addition of an eighth, most significant bit, which is ZERO.

Table 1 — Summary of assignments of Intermediate characters

<table>
<tr><td colspan="3" rowspan="2">First Intermediate character</td><td colspan="11">Second Intermediate character</td></tr>
<tr><td>2/0</td><td>2/1
2/2
2/3</td><td>2/4
2/5
2/6
2/7</td><td>2/8</td><td>2/9</td><td>2/10</td><td>2/11</td><td>2/12</td><td>2/13</td><td>2/14</td><td>2/15</td></tr>
<tr><td>Code position</td><td colspan="2">Category</td><td colspan="11"></td></tr>
<tr><td>2/0</td><td colspan="2">Announcers</td><td colspan="11"></td></tr>
<tr><td>2/1</td><td rowspan="3">Control functions</td><td>C0 set</td><td></td><td rowspan="3">Registration Authority use</td><td colspan="9"></td></tr>
<tr><td>2/2</td><td>C1 set</td><td></td><td colspan="9"></td></tr>
<tr><td>2/3</td><td>Single control functions</td><td></td><td></td><td colspan="4">94ⁿ set to</td><td></td><td colspan="3">96ⁿ set to</td></tr>
<tr><td>2/4</td><td>Graphic characters</td><td>Multiple-byte sets</td><td></td><td></td><td></td><td>G0 set</td><td>G1 set</td><td>G2 set</td><td>G3 set</td><td></td><td>G1 set</td><td>G2 set</td><td>G3 set</td></tr>
<tr><td>2/5</td><td colspan="2">Other coding systems</td><td></td><td>With standard return</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>No standard return</td></tr>
<tr><td>2/6</td><td colspan="2">Revision of Registration</td><td colspan="11"></td></tr>
<tr><td>2/7</td><td colspan="2"></td><td colspan="11"></td></tr>
<tr><td>2/8</td><td rowspan="4">94 Graphic characters</td><td>G0 set</td><td rowspan="4">DRCS</td><td rowspan="4">Registration Authority use</td><td rowspan="4" colspan="9"></td></tr>
<tr><td>2/9</td><td>G1 set</td></tr>
<tr><td>2/10</td><td>G2 set</td></tr>
<tr><td>2/11</td><td>G3 set</td></tr>
<tr><td>2/12</td><td colspan="2"></td><td colspan="11"></td></tr>
<tr><td>2/13</td><td rowspan="3">96 Graphic characters</td><td>G1 set</td><td rowspan="3">DRCS</td><td rowspan="3">Registration Authority use</td><td rowspan="3" colspan="9"></td></tr>
<tr><td>2/14</td><td>G2 set</td></tr>
<tr><td>2/15</td><td>G3 set</td></tr>
</table>

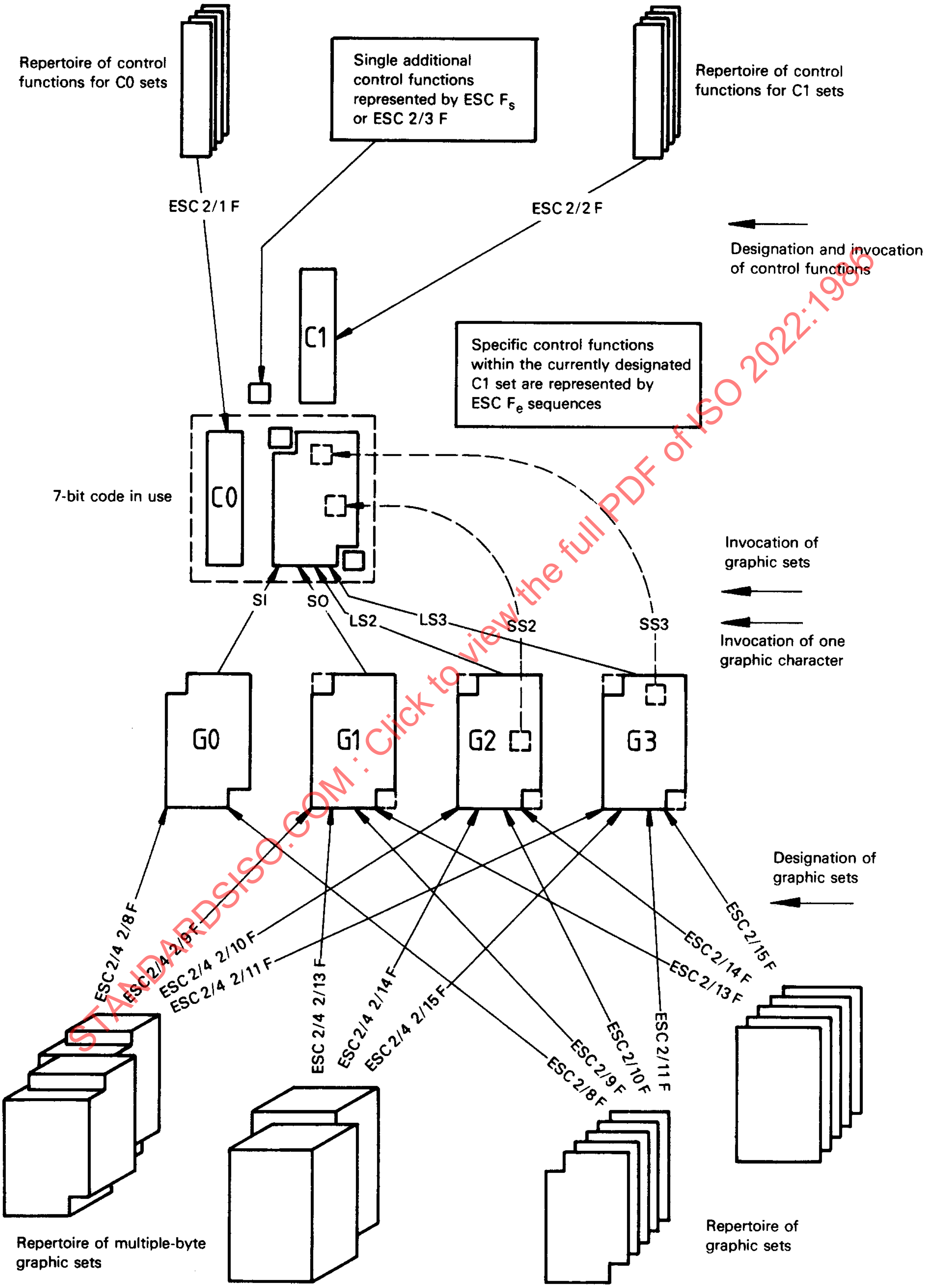

Repertoire of control functions for C0 sets
Single additional control functions represented by ESC F_s or ESC 2/3 F
Repertoire of control functions for C1 sets
ESC 2/1 F
ESC 2/2 F
Designation and invocation of control functions
C1
Specific control functions within the currently designated C1 set are represented by ESC F_e sequences
7-bit code in use
C0
Invocation of graphic sets
Invocation of one graphic character
SI
SO
LS2
LS3
SS2
SS3
G0
G1
G2
G3
Designation of graphic sets
ESC 2/4 2/8 F
ESC 2/4 2/9 F
ESC 2/4 2/10 F
ESC 2/4 2/11 F
ESC 2/4 2/13 F
ESC 2/4 2/14 F
ESC 2/4 2/15 F
ESC 2/8 F
ESC 2/9 F
ESC 2/10 F
ESC 2/11 F
ESC 2/13 F
ESC 2/14 F
ESC 2/15 F
Repertoire of multiple-byte graphic sets
Repertoire of graphic sets

Figure 7 — Code extension in a 7-bit environment (showing all shift functions)

Columns 08 to 15 of this array contain a further 128 code positions; the eighth bit of their coded representation is ONE.

Columns 08 and 09 are provided for control characters and columns 10 to 15 for graphic characters.

The control characters in columns 08 and 09 of an 8-bit code shall not include transmission control characters.

7.2 The family concept

In order to cope with the different needs of the various industries, fields of application or systems, this International Standard defines the concept of a family of 8-bit codes as follows :

a) a set of 32 additional control characters can be selected for columns 08 and 09;

b) a set of 94 or 96 additional graphic characters can be selected for columns 10 to 15. If a set of 94 graphic characters is invoked in columns 10 to 15, positions 10/0 and 15/15 shall not be used.

There are standard techniques for identifying selections of sets of control characters and graphic characters for 8-bit codes. These techniques are described below.

8 Use of code extension in an 8-bit code

The techniques of extending an 8-bit code described in this International Standard have been purposely made compatible with those used to extend the 7-bit code.

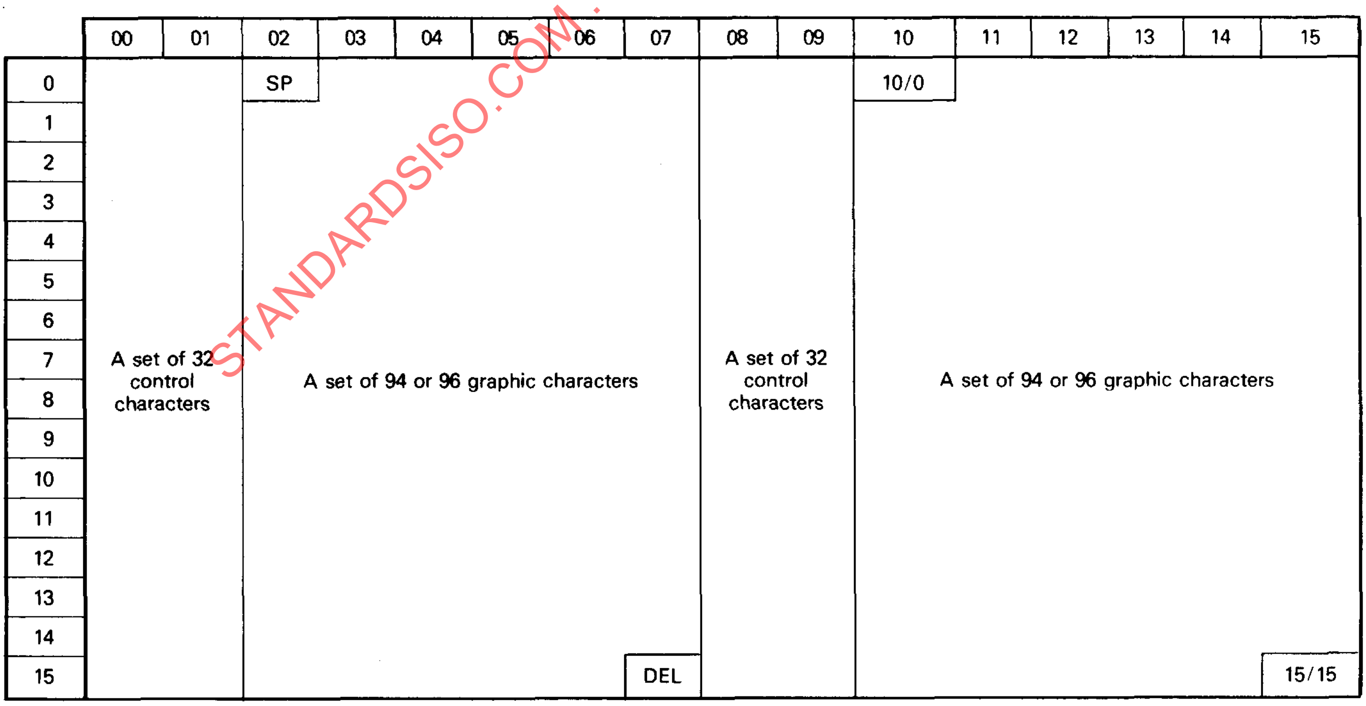
The character ESCAPE shall be used in an 8-bit code in exactly the same way as in the 7-bit code to construct escape sequences. The meanings of these sequences are not altered in an 8-bit code. All characters in columns 08 to 15 are excluded from assignment in escape sequences and any occurrences of them in an escape sequence are error conditions for which no standard recovery procedures are prescribed in this International Standard.

8.1 Elements of code extension in an 8-bit environment

These elements, shown in figure 2, are as follows :

Set	Description	Columns occupied
C0	32 control characters	00 to 01
C1	32 control characters	08 to 09
G0	94 graphic characters	02 to 07
G1	94 or 96 graphic characters	02 to 07 or 10 to 15
G2	94 or 96 graphic characters	02 to 07 or 10 to 15
G3	94 or 96 graphic characters	02 to 07 or 10 to 15

The C0 and C1 sets shall be designated and invoked by the same escape sequences as in the 7-bit environment (see 6.3.5 and 6.3.6). The G0, G1, G2 and G3 sets shall be designated by the same escape sequences as in the 7-bit environment (see 6.3.7 to 6.3.10).



00
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
SP
10/0
A set of 32 control characters
A set of 94 or 96 graphic characters
A set of 32 control characters
A set of 94 or 96 graphic characters
DEL
15/15

Figure 8 — 8-bit code structure

8.2 Extension of the graphic set by means of shift functions

The shift functions in this International Standard for use in an 8-bit environment are

LS0, LS1, LS1R, LS2, LS2R, LS3, LS3R, SS2 and SS3.

See annex B for the coded representation of these functions.

8.2.1 Use of locking-shift functions

In an 8-bit environment there are seven locking-shift functions used exclusively for graphic set extension. With the exception of LS0 which invokes only 94-character sets, each of the six others invokes an additional set of 94 or 96 graphic characters into columns 02 to 07 or into columns 10 to 15. The seven locking-shift functions are

Function		Set invoked	Columns affected
LOCKING-SHIFT ZERO	(LS0)	G0	02 to 07
LOCKING-SHIFT ONE	(LS1)	G1	02 to 07
LOCKING-SHIFT ONE RIGHT	(LS1R)	G1	10 to 15
LOCKING-SHIFT TWO	(LS2)	G2	02 to 07
LOCKING-SHIFT TWO RIGHT	(LS2R)	G2	10 to 15
LOCKING-SHIFT THREE	(LS3)	G3	02 to 07
LOCKING-SHIFT THREE RIGHT	(LS3R)	G3	10 to 15

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

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

a) those representing the control characters in columns 00, 01, 08 and 09;

b) those included in any escape sequence;

c) the one following SS2 or SS3.

The characters SPACE and DELETE shall occur at positions 02/0 and 07/15, respectively, if and only if a set of 94 graphic characters is invoked into columns 02 to 07. They shall not be assigned to any position 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 the locking-shift functions as specified in 6.4 (see also clause 9 and table 2).

8.2.2 Use of single-shift functions

Use of the single-shift functions in an 8-bit code is identical with their use in a 7-bit code (see 6.2.2). Only bit combinations from columns 02 to 07 are permitted to follow SS2 or SS3. Except for the situation described in 10.3, bit combinations from columns 10 to 15 shall not follow SS2 or SS3. The use of a single-shift function does not affect the current status established by one or more of the locking-shift functions.

8.3 Code extension by means of escape sequences

When an 8-bit code has been established in accordance with 8.1, code extension shall be achieved by means of escape sequences as described below.

8.3.1 Two-character escape sequences

Two-character escape sequences shall have the same structure as in the 7-bit environment (see 6.3.3.1).

ESC F_s sequences represent single additional control functions with the same meaning as in the 7-bit environment. (See the note to 6.3.3.1.)

The use of ESC F_e sequences in an 8-bit environment is contrary to the intention of this International Standard but should they occur in special circumstances (see table 2) their meaning is the same as in the 7-bit environment.

8.3.2 Three-character escape sequences

Three-character escape sequences shall have the same structure and meaning as in the 7-bit environment (see 6.3.3.2).

8.3.3 Escape sequences with four or more characters

These escape sequences shall have the same structure and meaning as in the 7-bit environment (see 6.3.3.3).

NOTE — The escape sequences to designate and invoke coding systems other than that specified in this International Standard (see 6.3.11), those to designate multiple-byte graphic sets (see 6.3.9) and those to designate dynamically redefinable character sets (see 6.3.10) have the same structure and meaning as in a 7-bit environment.

8.4 Sets of graphic characters with multiple-byte representation

In an 8-bit environment, as in a 7-bit environment, multiple-byte graphic character sets may be designated and invoked as G0, G1, G2 or G3 sets (see 6.3.9). A graphic character of such a multiple-byte set is represented by two or more bytes, all of which consist of bit combinations either from columns 02 to 07, or from columns 10 to 15, depending upon where the multiple-byte set has been invoked. Thus the eighth bit (b_8) of each byte in such a multiple-byte representation shall be uniformly either ZERO or ONE.

NOTES

1 If the eighth bit (b_8) of bytes in a given multiple-byte representation is different 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.

2 The transformation between 7-bit and 8-bit codes (see clause 10) is not affected by the occurrence of multiple-byte graphic characters sets.

8.5 Compatibility

An 8-bit code will be considered to be compatible with this International Standard if columns 00 to 07 comply with 6.1.5 a) or b) and

— columns 08 and 09 contain only control characters;

— columns 10 to 15 are used for graphic characters only.

In order to provide the code extension facilities of this International Standard, the character ESCAPE and the shift functions required must remain unaltered in their meanings and their coded representations (see annex B).

8.6 Pictorial representation of code extension in an 8-bit environment

Figure 9 summarizes, in a schematic form, the standard means of code extension available in an 8-bit environment.

9 Announcement of extension facilities used

9.1 General

At the beginning of an information interchange, it may be required to announce the code extension facilities used in the data which follows. If such announcement is to be embedded within the character coded information, one or more of the class of three-character escape sequences ESC 2/0 F shall be used. Subject to agreement between the interchanging parties, such an announcement sequence may be omitted. The Final character of the announcing sequence indicates the facilities used for representing graphic sets and some control sets in 7-bit and 8-bit environments. The Final characters used for this purpose are listed in table 2 together with a description of the facilities to be used and a pictorial representation where appropriate.

Escape sequences with 2/0 as the first Intermediate character and any further Intermediate characters are reserved for future standardization.

9.2 Restrictions

The announcers 4/1, 4/3 and 4/4 shall not be used in combination with the announcers 5/0, 5/2, 5/3, 5/4, 5/5, 5/6 and 5/7.

The announcers 4/12, 4/13 and 4/14 shall not be used together with any other announcer.

NOTES

1 In a 7-bit environment, data announced by a sequence ESC 2/0 4/4 have the same form as data announced by a sequence ESC 2/0 4/2. Both sequences are provided for those interchange situations in which it is required to differentiate in the 7-bit environment between data originating from two types of 8-bit environment, namely those having G1 in columns 2 to 7 and those having G1 in columns 10 to 15.

2 An example of the sequences which might be used in an 8-bit environment to announce the use of G0, G1 and G3 sets with locking shifts and G2 with a single shift, is as follows :

ESC 02/0 05/0
ESC 02/0 05/2
ESC 02/0 05/7
ESC 02/0 05/10

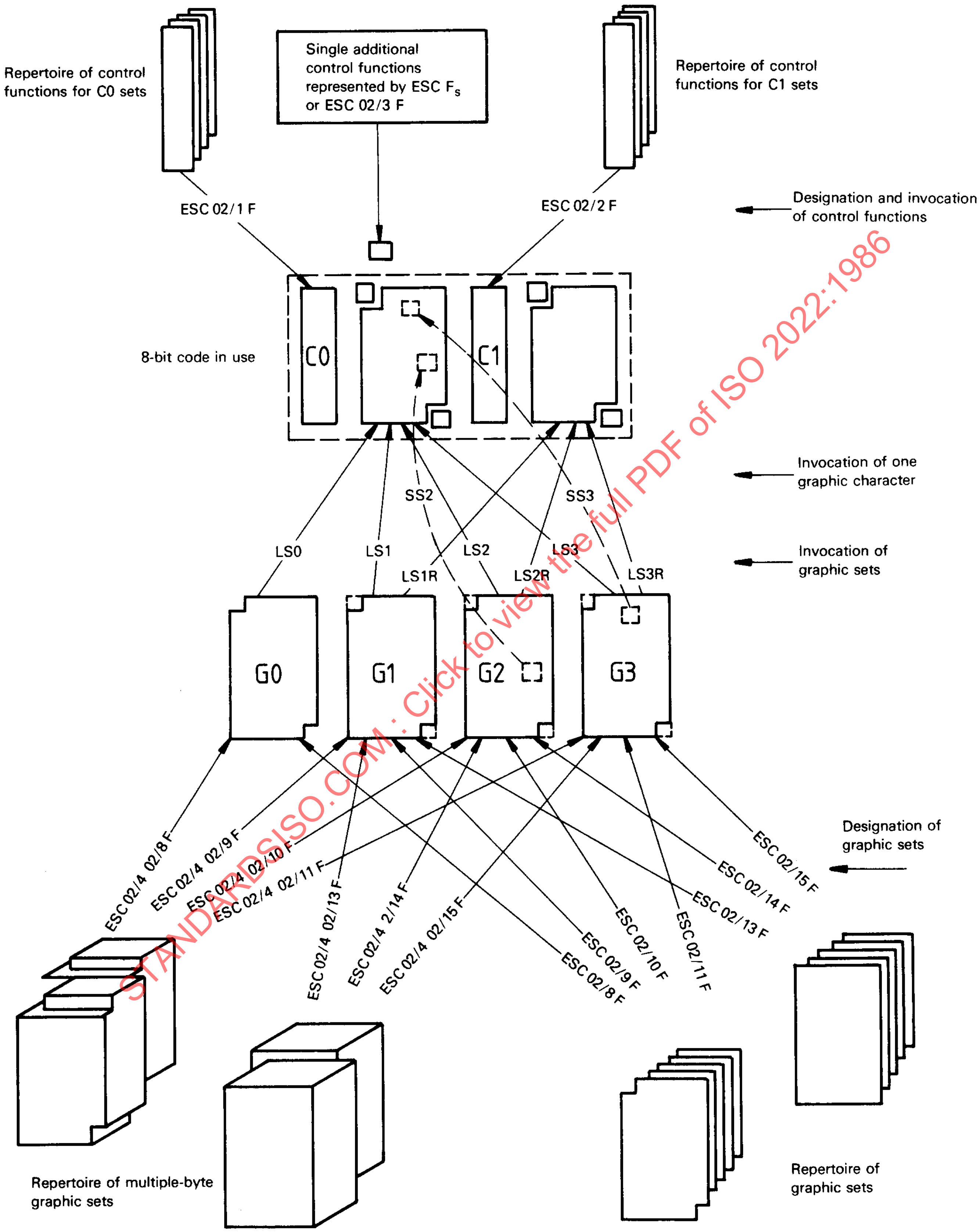

Repertoire of control functions for C0 sets
Single additional control functions represented by ESC F_s or ESC 02/3 F
Repertoire of control functions for C1 sets
ESC 02/1 F
ESC 02/2 F
Designation and invocation of control functions
8-bit code in use
C0
C1
Invocation of one graphic character
SS2
SS3
Invocation of graphic sets
LS0
LS1
LS2
LS3
LS1R
LS2R
LS3R
G0
G1
G2
G3
Designation of graphic sets
ESC 02/4 02/8 F
ESC 02/4 02/9 F
ESC 02/4 02/10 F
ESC 02/4 02/11 F
ESC 02/4 02/13 F
ESC 02/4 2/14 F
ESC 02/4 02/15 F
ESC 02/8 F
ESC 02/9 F
ESC 02/10 F
ESC 02/11 F
ESC 02/13 F
ESC 02/14 F
ESC 02/15 F
Repertoire of multiple-byte graphic sets
Repertoire of graphic sets

Figure 9 — Code extension in an 8-bit environment (showing all shift facilities)

Table 2 — Announcement of extension facilities

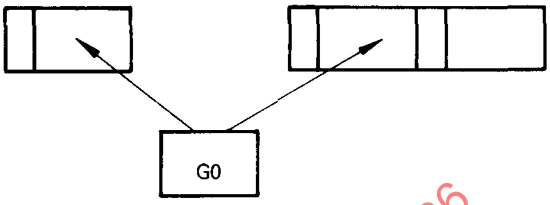
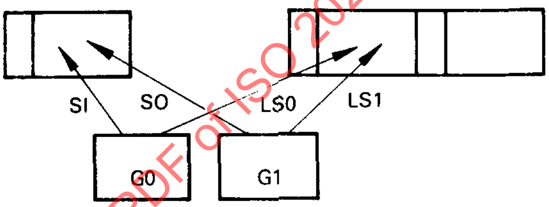
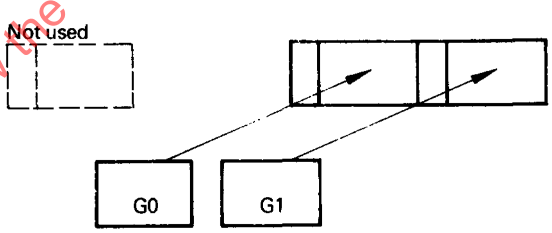
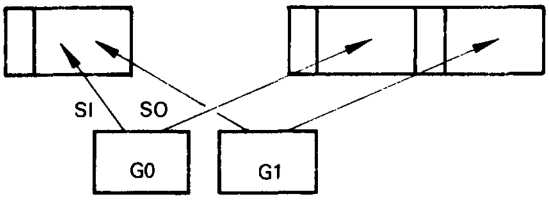
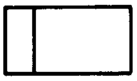
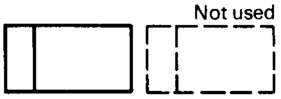
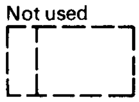
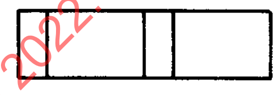
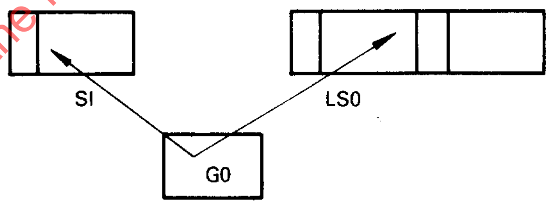
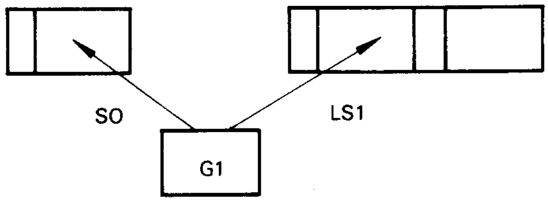
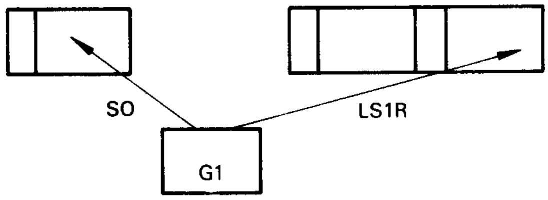
Final character	Facilities utilized	7-bit environment	8-bit environment
4/1	The G0 set shall be used. The escape sequence which designates this set also invokes it into columns 2 to 7. No locking-shift functions shall be used. In an 8-bit environment, columns 10 to 15 are not used. See 9.2.		
4/2	The G0 and G1 sets shall be used. In a 7-bit environment SI invokes G0 into columns 2 to 7 and SO invokes G1 into columns 2 to 7. In an 8-bit environment, LS0 invokes G0 and LS1 invokes G1 into columns 02 to 07, while columns 10 to 15 are not used.		
4/3	The G0 and G1 sets shall be used in an 8-bit environment only. The designating escape sequences also invoke the G0 and G1 sets into columns 02 to 07 and 10 to 15, respectively. No locking-shift functions shall be used. See 9.2.		
4/4	The G0 and G1 sets shall be used. In a 7-bit environment, SI invokes G0 and SO invokes G1 into columns 2 to 7. In an 8-bit environment, the designating escape sequences also invoke the G0 and G1 sets into columns 02 to 07 and 10 to 15, respectively; no locking-shift functions shall be used. See 9.2.		
4/5	Full preservation of shift functions is maintained when transforming data between 7-bit and 8-bit environments.	See 10.4	
4/6	The C1 set shall be used. In both a 7-bit and an 8-bit environment, each C1 control function shall be represented by an ESC F_e sequence.	See 6.3.3.1 and 8.3.1	
4/7	The C1 set shall be used. In a 7-bit environment, each C1 control function is represented by an ESC F_e sequence. In an 8-bit environment, each C1 control function is represented by a single bit combination from columns 08 and 09.	See 6.3.3.1	
4/8	All graphic character sets comprise 94 characters.	See 6.3.7	

Table 2 *(continued)*

Final character	Facilities utilized	7-bit environment	8-bit environment
4/9	The graphic character sets may comprise 94 and/or 96 characters.	See 6.3.7 and 6.3.8	
4/10	In a 7-bit or an 8-bit environment a 7-bit code is used.		
4/11	In an 8-bit environment an 8-bit code is used.		
4/12	Level 1 of ISO 4873 shall be used.		
4/13	Level 2 of ISO 4873 shall be used.		
4/14	Level 3 of ISO 4873 shall be used.		
5/0	In addition to any other category of graphic sets which may be used, the G0 set shall be used. It will be invoked by SI in a 7-bit environment and by LS0 in an 8-bit environment. See 9.2.		
5/2	In addition to any other category of graphic sets which may be used, the G1 set shall be used. It will be invoked by SO in a 7-bit environment and by LS1 in an 8-bit environment. See 9.2.		
5/3	In addition to any other category of graphic sets which may be used, the G1 set shall be used. It will be invoked by SO in a 7-bit environment and by LS1R in an 8-bit environment. See 9.2.		
5/4	In addition to any other category of graphic sets which may be used, the G2 set shall be used. It will be invoked by LS2 in both 7-bit and 8-bit environments. See 9.2.