International Standard 3309

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

Information processing systems — Data communication — High-level data link control procedures — Frame structure

Systèmes de traitement de l'information — Communication de données — Procédures de commande de liaison de données à haut niveau — Structure de trame

Third edition — 1984-10-01

UDC 681.327.8.01 **Ref. No. ISO 3309-1984 (E)**

Descriptors : data processing, teleprocessing, data transmission, communication procedure, high-level data link control, organization of data.

Price based on 6 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. Every 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 3309 was prepared by Technical Committee ISO/TC 97, *Information processing systems*.

The second edition of ISO 3309 was published in 1979. This third edition cancels and replaces the second edition, of which it constitutes a technical revision.

Printed in Switzerland

Information processing systems — Data communication — High-level data link control procedures — Frame structure

0 Introduction

This International Standard is one of a series to be used in the implementation of various applications with synchronous transmission facilities.

1 Scope and field of application

This International Standard specifies the frame structure for data communication systems using bit-oriented high-level data link control (HDLC) procedures. It defines the relative positions of the various components of the basic frame and the bit combination for the frame delimiting sequence (flag). The mechanism used to achieve bit pattern independence within the frame is also defined. In addition, two frame checking sequences (FCS) are specified; the rules for address field extension are defined; and the addressing conventions available are described.

Control field encodings and formats are defined in other International Standards.

2 Basic frame structure

In HDLC, all transmissions are in frames, and each frame consists of the following fields (transmission sequence left to right):

Flag	Address	Control	Information	FCS	Flag
01111110	8 bits	8 bits	*	16 or 32 bits	01111110

* An unspecified number of bits which in some cases may be a multiple of a particular character size; for example, an octet.

where

Flag = flag sequence

Address = data station address field

Control = control field

Information = information field

FCS = frame checking sequence field

Frames containing only control sequences form a special case where there is no information field. The format for these frames shall be

Flag	Address	Control	FCS	Flag
01111110	8 bits	8 bits	16 or 32 bits	01111110

3 Elements of the frame

3.1 Flag sequence

All frames shall start and end with the flag sequence. All data stations which are attached to the data link shall continuously hunt for this sequence. Thus, the flag is used for frame synchronization. A single flag may be used as both the closing flag for one frame and the opening flag for the next frame.

3.2 Address field

In command frames, the address shall identify the data station(s) for which the command is intended. In response frames, the address shall identify the data station from which the response originated.

3.3 Control field

The control field indicates the type of commands or responses, and contains sequence numbers, where appropriate. The control field shall be used

a) to convey a command to the addressed data station(s) to perform a particular orperation, or

b) to convey a response to such a command from the addressed data station.

3.4 Information field

Information may be any sequence of bits. In most cases it will be linked to a convenient character structure, for example octets, but, if required, it may be an unspecified number of bits and unrelated to a character structure.

3.5 Transparency

The transmitter shall examine the frame content between the two flag sequences including the address, control and FCS fields and shall insert a "0" bit after all sequences of 5 contiguous "1" bits (including the last 5 bits of the FCS) to ensure that a flag sequence is not simulated. The receiver shall examine the frame content and shall discard any "0" bit which directly follows 5 contiguous "1" bits.

3.6 Frame checking sequence (FCS) field

3.6.1 General

Two frame checking sequences are specified; a 16-bit frame checking sequence and a 32-bit frame checking sequence. The 16-bit frame checking sequence is normally used. The 32-bit frame checking sequence is for use by prior agreement in those cases that need a higher degree of protection than can be provided by the 16-bit frame checking sequence.

NOTES

1 If future applications show that other degrees of protection are needed, different numbers of bits in the FCS will be specified, but they will be an integral number of octets.

2 Explanatory notes on the implementation of the frame checking sequence are given in the annex.

3.6.2 16-bit frame checking sequence

The 16-bit FCS shall be the ones complement of the sum (modulo 2) of

a) the remainder of

$$x^k (x^{15} + x^{14} + x^{13} + x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1)$$

divided (modulo 2) by the generator polynomial

$$x^{16} + x^{12} + x^5 + 1,$$

where k is the number of bits in the frame existing between, but not including, the final bit of the opening flag and the first bit of the FCS, excluding bits inserted for transparency, and

b) the remainder of the division (modulo 2) by the generator polynomial

$$x^{16} + x^{12} + x^5 + 1$$

of the product of x^{16} by the content of the frame existing between, but not including, the final bit of the opening flag and the first bit of the FCS, excluding bits inserted for transparency.

As a typical implementation, at the transmitter, the initial content of the register of the device computing the remainder of the division is preset to all ones and is then modified by division by the generator polynomial (as described above) of the address, control and information fields; the ones complement of the resulting remainder is transmitted as the 16-bit FCS.

At the receiver, the initial content of the register of the device computing the remainder is preset to all ones. The final remainder after multiplication by x^{16} and then division (modulo 2) by the generator polynomial

$$x^{16} + x^{12} + x^5 + 1$$

of the serial incoming protected bits and the FCS will be

0001 1101 0000 1111 (x^{15} through x^0, respectively)

in the absence of transmission errors.

3.6.3 32-bit frame checking sequence

The 32-bit FCS shall be ones complements of the sum (modulo 2) of

a) the remainder of

$$x^k (x^{31} + x^{30} + x^{29} + x^{28} + x^{27} + x^{26} + x^{25} + x^{24} + x^{23} + x^{22} + x^{21} + x^{20} + x^{19} + x^{18} + x^{17} + x^{16} + x^{15} + x^{14} + x^{13} + x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1)$$

divided (modulo 2) by the generator polynomial

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1,$$

where k is the number of bits in the frame existing between, but not including, the final bit of the opening flag and the first bit of the FCS, excluding bits inserted for transparency, and

b) the remainder of the division (modulo 2) by the generator polynomial

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$$

of the product of x^{32} by the content of the frame existing between , but not including, the final bit of the opening flag and the first bit of the FCS, excluding bits inserted for transparency.

As a typical implementation, at the transmitter, the initial content of the register of the device computing the remainder of the division is preset to all ones and is then modified by division by the generator polynomial (as described above) of the address, control and information fields; the ones complement of the resulting remainder is transmitted as the 32-bit FCS.

At the receiver, the initial content of the register of the device computing the remainder is preset to all ones. The final remainder after multiplication by x^{32} and then division (modulo 2) by the generator polynomial

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$$

of the serial incoming protected bits and the FCS will be

1100 0111 0000 0100 1101 1101 0111 1011 (x^{31} through x^{0}, respectively)

in the absence of transmission errors.

3.7 Order of bit transmission

Addresses, commands, responses, and sequence numbers shall be transmitted low-order bit first (for example, the first bit of the sequence number that is transmitted shall have the weight 2^0).

The order of transmitting bits within the information field is not specified in this International Standard.

The FCS shall be transmitted to the line commencing with the coefficient of the highest term.

3.8 Inter-frame time fill

Inter-frame time fill shall be accomplished by transmitting either contiguous flags or seven to fourteen contiguous "1" bits or a combination of both.

Selection of the inter-frame time fill method depends on systems requirements.

3.9 Invalid frame

An invalid frame is defined as one that is not properly bounded by two flags or one which is too short (that is, shorter than 32 bits between flags when using the 16-bit FCS and shorter than 48 bits between flags when using the 32-bit FCS). Invalid frames shall be ignored. Thus, a frame which ends with an all "1" bit sequence of length equal to or greater than seven bits shall be ignored.

As an example, one method of aborting a frame would be to transmit 8 contiguous "1" bits.

4 Extensions

4.1 Extended address field

A single octet address field shall normally be used and all 256 combinations shall be available.

However, by prior agreement, the address field range can be extended by reserving the first transmitted bit (low-order) of each address octet which would then be set to binary zero to indicate that the following octet is an extension of the address field. The format of the extended octet(s) shall be the same as that of the first octet. Thus, the address field may be recursively extended. The last octet of an extended address field is indicated by setting the low-order bit to binary one.

When extension is used, the presence of a binary "1" in the first transmitted bit of the first address octet indicates that only one address octet is being used. The use of address extension thus restricts the range of single octet addresses to 128.

4.2 Extended control field

The control field may be extended by one or more octets. The extension methods and the bit patterns for the commands and responses are defined in related International Standard(s).

5 Addressing conventions

5.1 General

The following conventions shall apply in the assignment of addresses to data stations for which commands are intended.

5.2 All-station address

The address field bit pattern 11111111 is defined as the all-station address.

The all-station address shall only be used with command frames, and it shall instruct all receiving data stations to accept and action the associated command frame. Any response to a command with the all-station address shall contain the assigned individual address of the data station transmitting the response.

The all-station address may be used for all-station polling. When there is more than one receiving data station for which a command with the all-station address is intended, any responses from these data stations shall not interfere with one another.

NOTE — The mechanism used to avoid overlapping responses to a poll using the all-station address is not specified in this International Standard.

The all-station address may be used to determine the data link level identification (assigned address) of data station(s) when unknown; for example, in switched or reconfigured situations.

5.3 No-station address

The bit pattern 00000000 in the first octet of the extended or non-extended address field is defined as the no-station address.

The no-station address shall never be assigned to a data station.

The no-station address may be used for testing when it is intended that no data station react or respond to a frame containing the no-station address.

5.4 Group addresses

In addition to an individual assigned address, one or more data stations may be assigned one or more group addresses. A group address may be used, for example, for

a) transmitting a frame simultaneously to the assigned group of data stations

or

b) polling the assigned group of data stations.

Any address field bit pattern, except the all-station address, the no-station address and any individual addresses already assigned, may be assigned as a group address.

A group address may be used for group polling. When there is more than one data station for which a command with a group address is intended, any responses from these data stations shall not interfere with one another.

NOTE — The mechanism used to avoid overlapping responses to a poll using a group address is not specified in this International Standard.

Annex

Explanatory notes on the implementation of the frame checking sequence

(This annex does not form part of the standard.)

In order to permit the use of existing devices that are arranged to use a zero preset register, the following implementation may be used. (This example is given in terms of the 16-bit frame checking sequence.)

At the transmitter, generate the FCS sequence in the following manner while transmitting the elements of the frame unaltered onto the line:

a) preset the FCS register to zeros;

b) invert the first 16 bits (following the opening flag) before shifting them into the FCS register;

c) shift the remaining fields of the frame into the FCS register uninverted;

d) invert the contents of the FCS register (remainder) and shift onto the line as the FCS sequence.

At the receiver, operate the FCS checking register in the following manner while receiving (and storing) unaltered the elements of the frame as received from the line:

a) preset the FCS register to zeros;

b) invert the first 16 bits (following the opening flag) before shifting them into the FCS checking register;

c) shift the remaining elements of the frame, up to the beginning of the FCS, into the checking register uninverted;

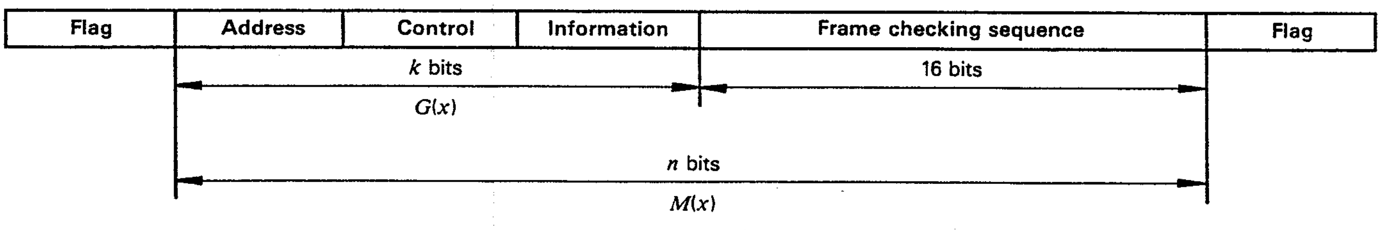
d) invert the FCS sequence before shifting into the checking register.

In the absence of errors, the FCS register will contain all zeros after the FCS is shifted in.

In the above, inversion of the first 16 bits is equivalent to a ones preset, and inversion of the FCS at the receiver causes the registers to go to the all zeros state.

The transmitter or the receiver can independently use the ones preset or the first 16-bit inversion. Also, the receiver can choose not to invert the FCS in which case it has to check for the unique nonzero remainder specified in 3.6.

It shall be realized that inversion of the FCS by the receiver requires a 16-bit storage delay before shifting received bits into the register. The receiver cannot anticipate the beginning of the FCS. Such storage, however, will normally take place naturally as the FCS checking function will need to differentiate the FCS from the data, and it will thus withhold 16 bits from the next function at all times.



The procedure for using the FCS is based on the assumptions that:

a) The k bits of data which are being checked by the FCS can be represented by a polynomial $G(x)$.

Example: $G(x) = x^5 + x^3 + 1$ represents 101001.

b) The address field, control field and information field (if it exists in the frame) are represented by the polynomial $G(x)$.

c) For the purpose of generating the FCS, the first bit following the opening flag is the most significant bit of $G(x)$ regardless of the actual representation of the address, control and information fields.

d) There exists a generator polynomial $P(x)$ of degree 16, having the form $P(x) = x^{16} + x^{12} + x^5 + 1$.