ECMA

EUROPEAN COMPUTER MANUFACTURERS ASSOCIATION

STANDARD ECMA-40

HDLC
FRAME STRUCTURE

3rd Edition – January 1980

ECMA

EUROPEAN COMPUTER MANUFACTURERS ASSOCIATION

STANDARD ECMA-40

HDLC
FRAME STRUCTURE

3rd Edition – January 1980

BRIEF HISTORY

In March 1970, ECMA received a proposal for a bit-oriented frame structure which was recognized to be an improvement over existing proposals. This new proposal was submitted by TC 9 to ISO/TC97/SC6 in 1970.

In June 1972, ISO/TC97/SC6 prepared a proposed Draft International Standard containing the bit-oriented frame structure and other features of high-level data link control (HDLC) and circulated it for voting as ISO DIS 3309.

The first edition of this Standard ECMA-40, in line with the ISO DIS was then published in December 1973.

Subsequently, improvements to the frame checking sequence in DIS 3309 prompted ECMA TC 9 to revise the Standard ECMA-40 to align it with the new ISO position. In this occasion, the Standard was editorially revised and information on special conditions and channel states were added. Though not present in ISO DIS 3309, these additions are in line with other ISO DIS on HDLC.

This 3rd edition has been prepared in order to bring the Standard editorially in line with the second edition of Standard ECMA-49, HDLC Elements of procedure, and the two new Standards ECMA-60, HDLC Unbalanced Class of Procedure and ECMA-61, HDLC Balanced Class of Procedure.

This 3rd Edition of ECMA-40 has been adopted by the General Assembly of ECMA on December 13, 1979.

THIS 3RD EDITION SUPERSEDES THE EDITION ISSUED IN SEPTEMBER 1976.

TABLE OF CONTENTS

1. SCOPE AND CONFORMANCE

This Standard ECMA-40 defines the frame structure for bit se-
quence independent data transmission. This structure is intend-
ed for use in High-Level Data Link Control (HDLC) applications.
This Standard defines in detail the relative positions of the
various components of the basic frame, the bit combination of
the frame delimiting sequence, the mechanism used to obtain
total transparency within the frame and the frame checking
sequence used.

Other standards define the details of the address and control
fields and the procedures used during the transmission.

Conformance with this Standard implies satisfying all the re-
quirements of sections 4,5,6,7 and 8 with either the normal
address field or the extended address field specified in 4.3,
with either a normal or an extended control field as specified
in 4.4 and with or without the optional information field spe-
cified in 4.5.

2. REFERENCES

ECMA-49   HDLC Elements of Procedure

ECMA-60   HDLC Unbalanced Class of Procedure

ECMA-61   HDLC Balanced Class of Procedure

3. DEFINITIONS

For the purpose of this Standard the following terms have the
meaning indicated.

3.1 Data Channel

A one-way digital transmission channel. It includes the trans-
mission media and intervening equipment involved in the trans-
fer of digital data in a given direction.

3.2 Frame

A sequence of bits, transmitted as a unit.

4. FRAME DEFINITION

4.1 Structure

The structure of a frame shall be as follows:

- Opening Flag (see 4.2)
- Address Field (see 4.3)
- Control Field (see 4.4)
- Information Field (see 4.5)
- Frame Check Sequence (FCS) (see 4.6)
- Closing Flag (see 4.7)

With the exception of the information field, which is option-
al, the above elements shall always be present in the frame
and in the order indicated. The information field, if pre-
sent, shall be in the position indicated.

The address field, control field, information field (when present) and the FCS constitute the frame content.

The two possible structures of a frame (with or without information field) and the position of the frame content are represented below.

OPENING FLAG	ADDRESS	CONTROL	INFORMATION	FCS	CLOSING FLAG
01111110	see 4.3	see 4.4	see 4.5	16 bits	01111110

frame content

- Frame containing an information field.

OPENING FLAG	ADDRESS	CONTROL	FCS	CLOSING FLAG
01111110	see 4.3	see 4.4	16 bits	01111110

frame content

- Frame not containing an information field.

4.2 Opening Flag

The opening flag shall consist of the following bit sequence:

ZERO ONE ONE ONE ONE ONE ONE ZERO

The flag sequence is used for frame synchronization. All stations participating in the data link shall continuously hunt for the flag sequence.

Since the opening and closing flag (see 4.7) are identical, contiguous frames can be separated by a single flag, which in this case acts as closing flag for one frame and opening flag for the following one.

A special transparency mechanism (see 6) is used to avoid simulated flag sequences within the frame content.

4.3 Address Field

The address field shall identify the Secondary involved in the interchange and hence the relationship between the Primary and the Secondary. The address structure is application-dependent and it is not identified in this Standard.

Two formats are possible for the address field: normal and extended. The choice between normal and extended address format is by agreement between sender and receiver.

4.3.1 Normal address field

In this format, the address field is eight bits long. All the 256 combinations shall be available for addressing.

- 3 -

4.3.2 Extended address field

In this format the address can occupy any integral number of octets. The first bit of each octet shall be set to ZERO to indicate that the following octet is an address octet, or to ONE to indicate that the octet is the last (or only) octet of the address field. For this reason, the use of address extension restricts the range of single-octet addresses to 128.

4.4 Control Field

The control field is used by the Primary to send commands to the Secondary, and by the Secondary to respond to the Primary. The sequence numbers, where used, are contained in the control field.

The control field can be one (normal) or two (extended) octets long. The coding of the control field, either normal or extended, is defined in Standard ECMA-49.

4.5 Information Field

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

The maximum number of bits permitted in the information field is application and/or system dependent, and is not specified in this Standard. The presence of the information field is optional.

4.6 Frame Check Sequence (FCS)

All frames shall include a 16-bit frame check sequence (FCS) just prior to the closing flag for error detection purpose.

The FCS is calculated on all k bits of the transmitted frame i.e. the address field, the control field and the information field, if present, excluding the bits inserted for transparency, as defined in 6. Denoting the k bits by a_{k-1} ... ... a_0, where the ith bit transmitted is assigned the index $k-i$, this part of the contents can be represented by:

$$P_{k-1}(X) = a_{k-1}X^{k-1} + a_{k-2}X^{k-2} \ldots + a_1X + a_0$$

The generating polynomial shall be:

$$G(X) = X^{16} + X^{12} + X^5 + 1$$

Generation and checking of the FCS are defined in 4.6.1 and 4.6.2. Annex A gives additional details on the calculations involved.

4.6.1 FCS generation

At the transmitter the FCS is calculated as the sum (modulo 2) of the following three terms:

A) - The remainder of the division (modulo 2) of

$$I(X) = X^k (X^{15} + X^{14} + ... + X + 1) \text{ by } G(X)$$

B) - The remainder of the division (modulo 2) of

$$X^{16} P_{k-1} (X) \text{ by } G(X)$$

C)

$$X^{15} + X^{14} + ... + X + 1$$

This FCS is transmitted as a 16-bit sequence with higher order coefficient first, so that the polynomial transmitted is:

$$M(X) = X^{16} P_{k-1} (X) + FCS(X)$$

If errors occur on the link, the polynomial received will be:

$$M'(X) = M(X) + E(X)$$

where $E(X)$ represents the error polynomial.

4.6.2 FCS checking

At the receiver, the received frame is checked to detect transmission errors as follows:

The sum (modulo 2) of the following two terms:

D) - The remainder of the division (modulo 2) of

$$X^{16} I(X) = X^{k+16} (X^{15} + X^{14} + ... + X + 1) \text{ by } G(X)$$

E) - The remainder of the division (modulo 2) of

$$X^{16} M'(X) \text{ by } G(X)$$

shall be, in the absence of detectable errors:

$$X^{12} + X^{11} + X^{10} + X^8 + X^3 + X^2 + X + 1$$

4.7 Closing Flag

The closing flag, identical to the opening flag, shall con-
sist of the following bit sequence:

ZERO ONE ONE ONE ONE ONE ONE ZERO

This closing flag may act as opening flag of the following
frame.

5. ORDER OF TRANSMISSION

The elements of a frame shall be transmitted as follows:

- opening flag
- address field
- control field
- information field, when present
- frame check sequence
- closing flag

Address and control fields shall be transmitted least signi-
ficant bit first. The frame check sequence shall be trans-
mitted higher coefficient bit first, as defined in 4.6.1.

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

6. FRAME CONTENT TRANSPARENCY

The transmitter shall examine the frame content and insert a
ZERO bit after all sequences of 5 adjacent ONE bits (includ-
ing the last 5 bits of the FCS) to ensure that a flag, abort
or idle sequence is not simulated.

The receiver shall examine the frame content and shall dis-
card any ZERO bit which directly follows 5 adjacent ONE bits.

7. SPECIAL CONDITIONS

7.1 Inter Frame Time Fill

Contiguous flag sequences may be used to fill time between
frames.

7.2 Idle Sequence

The idle condition on the receive channel of the data cir-
cuit shall be indicated to a receiver by means of 15 or
more contiguous ONE bits.

7.3 Abort Sequence

Aborting the transmission of a frame is accomplished by
transmitting an abort sequence, consisting of at least se-
ven contiguous ONE bits with no inserted ZEROs.

7.4 Invalid Frame

A frame is considered invalid and discarded if:

- it is terminated by an abort sequence;
- it contains less than 32 bits (excluding the ZERO bits inserted for transparency);
- it is terminated by an invalid FCS.

8. DATA CHANNEL STATES

8.1 Active Data Channel State

A data channel is considered to enter the active state when a receiver detects a flag sequence. It will leave that state when 15 contiguous ONE bits are detected.

8.2 Idle Data Channel State

A data channel is considered to enter the idle state when a receiver detects 15 contiguous ONE bits. It will leave that state when a flag sequence is detected.

ANNEX A

Additional Information on the Frame Check Sequence

A.1 General

An algebraic notation based on modulo 2 arithmetic is used
to describe the FCS generating and checking process. In this
notation bit sequences are represented by means of polynomials.
For example, bit sequence **1 0 1 0 0 1 0 0** is represented by
the polynomial

$$f(X) = X^7 + X^5 + X^2$$

Note that the leading bits at the left-hand side correspond
to the high order coefficients of the polynomial.

It will be supposed that high order coefficients correspond
to bits, which are transmitted first.

A.2 Notations Used

If

- $P_{k-1}(X)$ represents the **k** bits of address field, control
  field and (optional) information field,

- **FCS(X)** represents the 16-bit frame check sequence, which
  is transmitted adjacent to the above mentioned **k** bits, with
  the higher order coefficient first,

- **M(X)** represents the frame content, including the FCS, and
  consisting of **k + 16** bits,

then

$$M(X) = X^{16} P_{k-1}(X) + FCS(X).$$

Note that the multiplication of $P_{k-1}(X)$ **by** X^{16} just creates
space for the addition of the frame check sequence **FCS(X)**.

If

- **M'(X)** represents the frame content, which is received,

then

$$M'(X) = M(X) + E(X)$$

(1)

where **E(X)** denotes the polynomial corresponding to the bits in error.

A.3 Mathematical Equivalence of Inverting a Sequence Bit by Bit

The polynomial expression of a bit sequence of n bits is:

$$R(X) = \sum_{i=0}^{n-1} a_i X^i$$

Inverting bit by bit or complementing this bit sequence is equivalent to replacing $\overline{a_i} = 1 + a_i$ (modulo 2). Thus

$$\overline{R(X)} = \sum_{i=0}^{n-1} \overline{a_i} X^i = \sum_{i=0}^{n-1} (1+a_i) X^i = \sum_{i=0}^{n-1} X^i + \sum_{i=0}^{n-1} a_i X^i$$

where

$$\sum_{i=0}^{n-1} X^i = X^{n-1} + X^{n-2} + \ldots + 1$$

A.4 FCS Generation

- Term A of the sum mentioned in 4.6.1 is necessary to protect against addition of leading ZERO bits or leading flags corrupted into ZERO bits. See A.5, term D.

NOTE A.1

When the FCS is generated in a cyclic shift register, the first element is performed by either pre-loading the shift register with all ONES or by inverting the 16 high order coefficient bits of $X^{16} P_{k-1}(X)$ before feeding them into the shift register.

- Term B of the sum is the modulo 2 division by the generating polynomial **G(X)**.

- Term C of the sum is intended to cause a non-zero, unique remainder at the receiver checking. It causes the bit by bit inverting of the sum of term A and term B as shown in A.3.

  *In algebraic notation:

  In the FCS generation term A plus term B is the remainder of the division:

$$\frac{X^{16} P_{k-1}(X) + X^k \sum_{n=0}^{15} X^n}{G(X)}$$

giving the result:

$$Q(X) + \frac{R(X)}{G(X)}$$

where: $G(X) = X^{16} + X^{12} + X^5 + 1$

$Q(X)$ is the quotient

$R(X)$ is the remainder

that is:

term A + term B = $R(X)$

term A + term B + term C = $\overline{R(X)}$

The algebraic expression of FCS is:

$$FCS(X) = \overline{R(X)} = G(X)\,Q(X) + X^{16}P_{k-1}(X) + (1+X^k)\sum_{n=0}^{15} X^n$$

The bit sequence to be transmitted is given by:

$$M(X) = X^{16}P_{k-1}(X) + FCS \qquad\qquad (2)$$

or

$$M(X) = G(X)\,Q(X) + (1+X^k)\sum_{n=0}^{15} X^n$$

A.5 FCS Checking

- Term D is calculated in the same way as term A, so they are balanced. If, however, bits (especially ZERO bits) are inserted in front of $M(X)$ there is a change in the check-sum and the error can be detected.

NOTE A.2

When a cyclic shift register is used to perform the check, the first element is performed by either pre-loading the cyclic shift register with all ONES or by inverting the 16 higher order coefficients of $X^{16}M'(X)$ prior to feeding those to the cyclic shifts register.

- Term E is obtained by multiplying the polynomial by X^{16} and then dividing modulo 2 by the generating polynomial.

   *In algebraic notation

   The above mentioned divisions can be presented as follows:

$$\frac{\left(M'(X) + X^k \sum_{n=0}^{15} X^n\right) X^{16}}{G(X)}$$

which, using identities (1) and (2) above, can be re-written as:

$$X^{16} Q(X) + \frac{X^{16}\left(E(X) + \sum_{n=0}^{15} X^n\right)}{G(X)}$$

In absence of errors, the unique remainder is the remainder of the division

$$\frac{X^{16} \sum_{n=0}^{15} X^n}{G(X)}$$

This results into $X^{12} + X^{11} + X^{10} + X^8 + X^3 + X^2 + X + 1$ which corresponds to the bit sequence:

$$0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1$$

ANNEX B

Differences Between Editions

B.1 Differences Between the First and Second Edition

Beside a complete re-editing, with better presentation and distribution of the original contents, there are two main differences between the first and second edition of this Standard ECMA-40.

The first one is in the domain of the Frame Check Sequence (FCS). While the generating polynomial remains the same, the new checking method is based on a new generation and checking of the FCS, in order to protect against errors that would be undetected if flag sequences are garbled. The second edition contains a detailed explanation on the generation and checking of the FCS and an Appendix that gives a complete background information on the FCS calculation.

The second difference consists in the addition of two sections, section 7, Special Conditions, and section 8, Data Channel States. Section 7 contains information on Inter Frame Time Fill, Idle Sequence, Abort Sequence and Invalid Frame. Section 8 contains definitions on Active and Idle Data Channel States. It was felt that the information contained in these two sections was more related to the Frame Structure than to the Elements of Procedure Standard.

B.2 Differences Between the Second and Third Edition

The differences between the second and third edition are purely editorial.