INTERNATIONAL STANDARD

ISO
8372

First edition
1987-08-15

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

Information processing — Modes of operation for a 64-bit block cipher algorithm

Traitement de l'information — Modes opératoires d'un algorithme de chiffrement par blocs de 64 bits

Reference number
ISO 8372 : 1987 (E)

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 8372 was prepared by Technical Committee ISO/TC 97, *Information processing systems.*

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

Information processing — Modes of operation for a 64-bit block cipher algorithm

1 Scope and field of application

This International Standard describes four modes of operation for any 64-bit block cipher algorithm using a secret key.

NOTE — The annex, which does not form part of this International Standard, contains comments on the properties of each mode.

This International Standard establishes four defined modes of operation so that in any application of a 64-bit block cipher (for example data transmission, data storage authentication) this International Standard will provide a useful reference for the specification of the mode of operation, the formation of the starting variable, and the values of parameters (as appropriate).

NOTE — For the Cipher Feedback (CFB) mode of operation (see clause 7), two parameters, j and k, are defined. For the Output Feedback (CFB) mode of operation (see clause 8), one parameter, j, is defined. When one of these modes of operation is used the parameter value(s) needs to be chosen and used by all communicating parties.

2 Reference

ANSI X3.92-1981, *Data Encryption Algorithm*.

3 Definitions

3.1 plaintext: Unenciphered information.

3.2 cipher text: Enciphered information.

3.3 block chaining: The encipherment of information such that each block of ciphertext is cryptographically dependent upon the preceding ciphertext block.

3.4 initializing value (IV): Value used in defining the starting point of an encipherment process.

3.5 starting variable (SV): Variable derived from the initializing value and used in defining the starting point of the modes of operation.

NOTE — The method of deriving the starting variable from the initializing value is not defined in this International Standard. It needs to be described in any application of the modes of operation.

3.6 cryptographic synchronization: The co-ordination of the encipherment and decipherment process.

4 Notation

For the purposes of this International Standard the functional relation defined by the block encipherment algorithm is written

$$C = eK(P)$$

where

P is the plaintext block;

C is the ciphertext block;

K is the key.

The expression eK is the operation of encipherment using the key K.

The corresponding decipherment function is written

$$P = dK(C)$$

A variable, such as P and C above, denoted by a capital letter, represents a one-dimensional array of bits, for example:

$$A = \{a_1, a_2, \ldots, a_m\} \qquad B = \{b_1, b_2, \ldots, b_m\}$$

i.e. arrays of m bits, numbered from 1 to m.

The operation of addition, modulo 2, also known as the exclusive 'or' function is shown by the symbol $\oplus$. The operation applied to arrays such as A and B is defined as

$$A \oplus B = \{a_1 \oplus b_1, a_2 \oplus b_2, \ldots, a_m \oplus b_m\}$$

The operation of selecting the left-most j bits of A to generate a j bit array is written

$$A \sim j = \{a_1, a_2, \ldots, a_j\}$$

This operation is defined only when $j < m$, where m is the number of bits in A.

A "shift function" S_k is defined as follows.

Given an m-bit variable X and a k-bit variable F where $k < m$, the effect of a shift function $S_k(X|F)$ produces the m-bit variable

$$S_k(X|F) = \{x_{k+1}, x_{k+2}, \ldots, x_m, f_1, f_2, \ldots, f_k\}$$

The effect is to shift the bits of array X left by k places, discarding $x_1 \ldots x_k$ and to place the array F in the rightmost k places of X.

A special case of this function is used which begins with the k-bit variable $I(k)$ of successive 1 bits and shifts the variable C of j bits into it, where $j \leqslant k$.

The result is

$$S_j(I(k) \mid C) = \{1, 1, \ldots, 1, c_1, c_2, \ldots, c_j\}$$

where there are $k - j$ "ones" on the left of the resultant array.

5 Electronic Codebook (ECB) mode

Given a plaintext block P of 64 bits, the encipherment algorithm produces a ciphertext block C of 64 bits, i.e.:

$$C = eK(P)$$

The decipherment algorithm produces

$$P = dK(C)$$

This mode of using the encipherment algorithm is known as "electronic codebook".

6 Cipher Block Chaining (CBC) mode

The variables employed for the CBC mode of encipherment are

a) a sequence of n plaintext blocks $P_1, P_2, \ldots, P_n$, each of 64 bits;

b) a key K;

c) a starting variable SV of 64 bits;

d) the resultant sequence of n ciphertext blocks $C_1, C_2, \ldots, C_n$, each of 64 bits.

NOTE — The method of forming SV is not described in this International Standard.

The CBC mode of encipherment is described as follows:

Encipherment of the first plaintext variable:

$$C_1 = eK(P_1 \oplus SV) \qquad \ldots (1)$$

subsequently,

$$C_i = eK(P_i \oplus C_{i-1}) \quad \text{for } i = 2, 3, \ldots, n \qquad \ldots (2)$$

This procedure is illustrated in the upper part of figure 1. The starting variable SV is used in the generation of the first ciphertext output.

Subsequently, the ciphertext is added, modulo 2, to the next plaintext before encipherment.

The CBC mode of decipherment is described as follows:

Decipherment of the first ciphertext variable:

$$P_1 = dK(C_1) \oplus SV \qquad \ldots (3)$$

subsequently,

$$P_i = dK(C_i) \oplus C_{i-1} \quad \text{for } i = 2, 3, \ldots, n \qquad \ldots (4)$$

This procedure is illustrated in the lower part of figure 1.

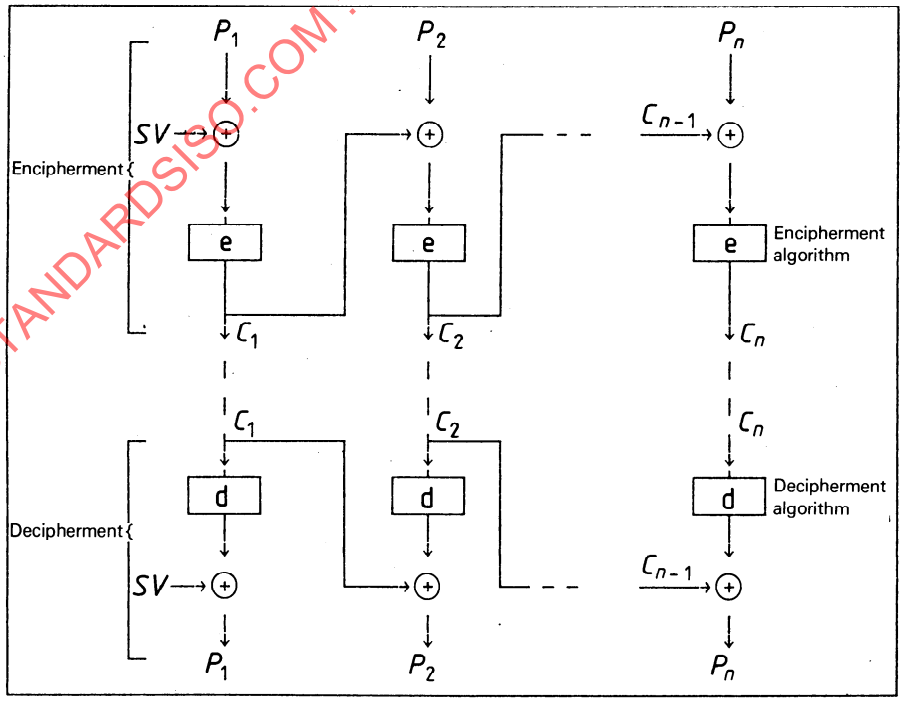

Figure 1 — Cipher Block Chaining (CBC) mode of operation

7 Cipher Feedback (CFB) mode

7.1 Two parameters define a CFB mode of operation

a) the size of feedback variable, k, where $1 \leqslant k \leqslant 64$;

b) the size of plaintext variable, j, where $1 \leqslant j \leqslant k$.

The variables employed for the CFB mode of operation are

a) The input variables:

1) a sequence of n plaintext variables $P_1, P_2, \ldots, P_n$, each of j bits;

2) a key K;

3) a starting variable SV of 64 bits.

b) The intermediate results:

1) a sequence of n algorithm input variables $X_1, X_2, \ldots, X_n$, each of 64 bits;

2) a sequence of n algorithm output variables $Y_1, Y_2, \ldots, Y_n$, each of 64 bits;

3) a sequence of n variables $E_1, E_2, \ldots, E_n$, each of j bits;

4) a sequence of n feedback variables $F_1, F_2, \ldots, F_n$, each of k bits.

c) The output variables, i.e. a sequence of n ciphertext variables $C_1, C_2, \ldots, C_n$, each of j bits.

NOTE — The method of forming SV is not described in this International Standard.

The variable X is set to its initial value

$$X_1 = SV \qquad \ldots (5)$$

7.2 The operation of enciphering each plaintext block employs the following five steps:

a) use of encipherment algorithm, $Y_i = eK(X_i)$; ... (6)

b) selection of leftmost j bits, $E_i = Y_i \sim j$; ... (7)

c) generation of ciphertext block, $C_i = P_i \oplus E_i$; ... (8)

d) generation of feedback block, $F_i = S_j(I(k)|C_i)$; ... (9)

e) shift function on X, $X_{i+1} = S_k(X_i|F_i)$. ... (10)

These steps are repeated for $i = 1, 2, \ldots, n$, ending with equation (8) on the last cycle. The procedure is illustrated on the left side of figure 2. The leftmost j bits of the output Y of the encipherment algorithm are used to encipher the j-bit plaintext block by modulo 2 addition. The remaining bits of Y are discarded. The bits of the plaintext and ciphertext blocks are numbered from 1 to j.

The ciphertext block is augmented by placing $k - j$ "ones" in its leftmost bit positions to become F, a k-bit array, then the bits of the array X are shifted left by k places and the array F is inserted in the rightmost k places, to produce the new value of X. In this shift operation, the leftmost k bits of X are discarded. The initial value of the array X is the starting variable (SV).

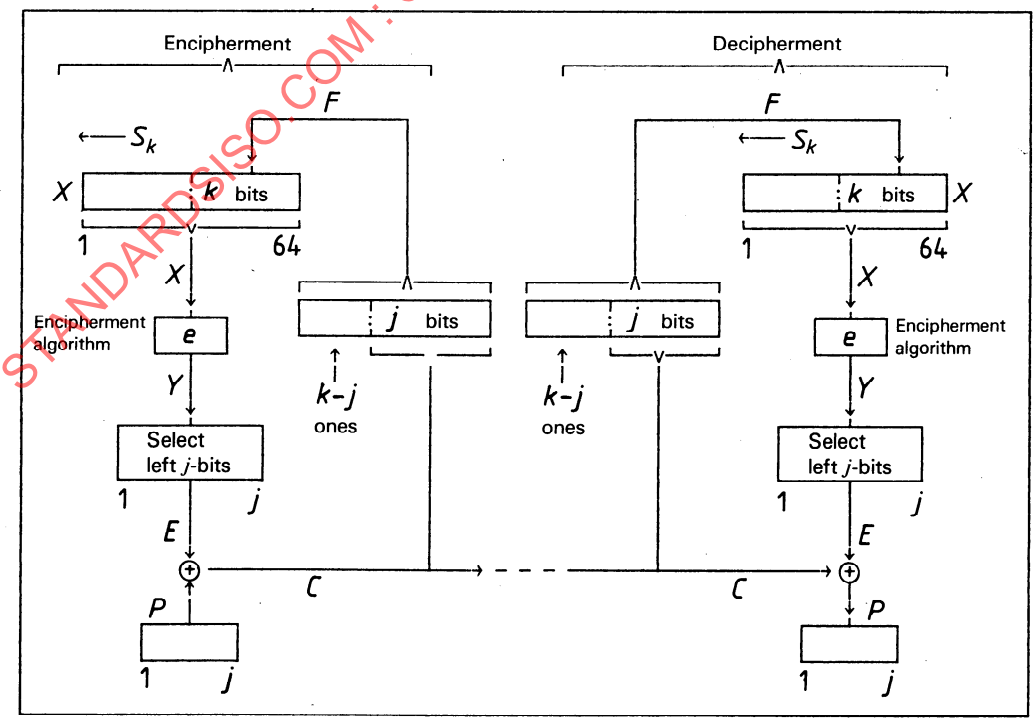

Figure 2 — Cipher Feedback (CFB) mode of operation

7.3 The variables employed for decipherment are the same as those employed for encipherment. The variable X is set to its initial value $X_1 = SV$.

The operation of deciphering each ciphertext block employs the following five steps:

a) use of encipherment algorithm, $Y_i = eK(X_i)$; . . . (11)

b) selection of leftmost j bits, $E_i = Y_i \sim j$; . . . (12)

c) generation of plaintext block, $P_i = C_i \oplus E_i$; . . . (13)

d) generation of feedback block, $F_i = S_j(I(k)|C_i)$; . . . (14)

e) shift function on X, $X_{i+1} = S_k(X_i|F_i)$. . . . (15)

These steps are repeated for $i = 1, 2, \ldots, n$, ending with equation (13) on the last cycle. The procedure is illustrated on the right side of figure 2. The leftmost j bits of the output Y of the encipherment algorithm are used to decipher the j-bit ciphertext block by modulo 2 addition. The remaining bits of Y are discarded. The plaintext and ciphertext blocks have bits numbered from 1 to j.

The ciphertext block is augmented by placing $k - j$ "ones" in its leftmost bit positions to become F, a k-bit array, then the bits of the array X are shifted left by k places and the array F is inserted in the rightmost k places to produce the new value of X. In this shift operation, the leftmost k bits of X are discarded. The initial value of the array X is the starting variable (SV).

7.4 It is recommended that CFB should be used with equal values of j and k.

In this recommended form ($j = k$) the equations (9) and (14) can be written

$F_i = C_i$ (case $j = k$)

8 Output Feedback (OFB) mode

8.1 One parameter defines an OFB mode of operation, i.e. the size of plaintext variable j where $1 \leqslant j \leqslant 64$.

The variables employed for the OFB mode of operation are

a) The input variables

1) a sequence of n plaintext blocks $P_1, P_2, \ldots, P_n$, each of j bits;

2) a key K;

3) a starting variable SV of 64 bits.

b) The intermediate results:

1) a sequence of n algorithm input variables $X_1, X_2, \ldots, X_n$, each of 64 bits;

2) a sequence of n algorithm output variables $Y_1, Y_2, \ldots, Y_n$, each of 64 bits;

3) a sequence of n variables $E_1, E_2, \ldots, E_n$, each of j bits.

c) The output variables, i.e. a sequence of n ciphertext variables $C_1, C_2, \ldots, C_n$, each of j bits.

NOTE — The method of forming SV is not described in this International Standard.

The variable X is set to its initial value

$X_1 = SV$ . . . (16)

8.2 The operation of enciphering each plaintext block employs the following four steps:

a) use of encipherment algorithm, $Y_i = eK(X_i)$; . . . (17)

b) selection of leftmost j bits, $E_i = Y_i \sim j$; . . . (18)

c) generation of ciphertext block, $C_i = P_i \oplus E_i$; . . . (19)

d) feedback operation, $X_{i+1} = Y_i$. . . . (20)

These steps are repeated for $i = 1, 2, \ldots, n$, ending with equation (19) on the last cycle. The procedure is illustrated on the left side of figure 3. The result of each use of the encipherment algorithm, which is Y_i, is used to feed back and become the next value of X, namely X_{i+1}. The leftmost j bits of Y_i are used to encipher the input block.

8.3 The variables employed for decipherment are the same as those employed for encipherment. The variable X is set to its initial value $X_1 = SV$.

The operation of deciphering each ciphertext block employs the following four steps:

a) use of encipherment algorithm, $Y_i = eK(X_i)$; . . . (21)

b) selection of leftmost j bits, $E_i = Y_i \sim j$; . . . (22)

c) generation of plaintext block, $P_i = C_i \oplus E_i$; . . . (23)

d) feedback operation, $X_{i+1} = Y_i$. . . . (24)

These steps are repeated for $i = 1, 2, \ldots, n$, ending with equation (23) on the last cycle. The procedure is illustrated in the right side of figure 3. The values of variables X_i and Y_i are the same as those used for encipherment; only equation (23) is different.

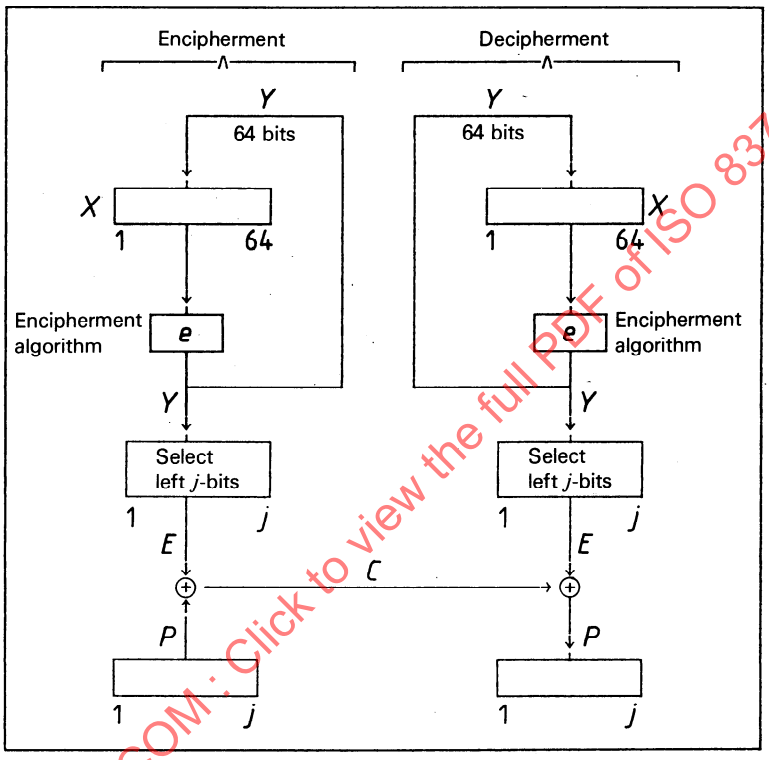

Figure 3 — Output Feedback (OFB) mode of operation

Annex

Properties of the modes of operation

(This annex contains comments on the properties of the four modes of operation described in this standard and is not an integral part of the body of the standard.)

A.1 Properties of the Electronic Codebook (ECB) mode of operation

Messages that carry information between computers, or people, may have repetitions or commonly used sequences. In ECB mode, identical plaintext produces (for the same key) identical ciphertext variables. This characteristic makes ECB unsuitable for general use. The use of ECB may be specified in future standards for those purposes where the repetition characteristic is acceptable.

If block boundaries are lost between encipherment and decipherment (for example a bit slip), synchronization between the encryption and decryption operations will be lost until correct block boundaries are re-established. The results of all decipherment operations will be incorrect.

A.2 Properties of the Cipher Block Chaining (CBC) mode of operation

The CBC mode produces the same ciphertext whenever the same plaintext is enciphered using the same key and initialising value. Users who are concerned about this characteristic should devise some way of changing the start of the plaintext, the key or the starting variable. One possibility is to incorporate a unique identifier (for example an incremented counter) at the beginning of each CBC message. Another, which may be used when encrypting records whose size should not be increased, is to use some value as the initialising value which can be computed from the record without knowing its contents (for example the number of the block which contains it in random access storage).

Since the CBC mode is a block method of encipherment, it needs to operate on complete data blocks of 64 bits. Blocks of less than 64 bits require special handling.

In the CBC mode, one or more bit errors within a single ciphertext block will affect the decipherment of two blocks (the block in which the error occurs and the succeeding block). If the errors occur in the ith ciphertext block, each bit of the ith plaintext block will have an average error rate of 50 %. The $(i + 1)$th plaintext block will have only those bits in error that correspond directly to the ciphertext bits in error.

If block boundaries are lost between encipherment and decipherment (for example a bit slip), synchronization between the encryption and decryption operations will be lost until correct block boundaries are re-established. The results of all decipherment operations will be incorrect.

A.3 Properties of the Cipher Feedback (CFB) mode of operation

In the CFB mode, errors in any j-bit unit of ciphertext will affect the decipherment of the garbled ciphertext and also the decipherment of succeeding ciphertext until the bits in error have been shifted out of the CFB input block. The first affected j-bit unit of plaintext will be garbled in exactly those places where the ciphertext is in error. Succeeding deciphered plaintext will have an average error rate of 50 % until all errors have been shifted out of the input block. Assuming no additional errors are encountered during this time, the correct plaintext will then be obtained. This characteristic is referred to as "limited error extension".

If j-bit boundaries are lost during decryption, cryptographic synchronization will be lost until cryptographic initialisation is performed or until 64 bits after the j-bit boundaries have been re-established.

The encipherment and decipherment processes in the CFB mode both use the encipherment form of the algorithm.

A.4 Properties of the Output Feedback (OFB) mode of operation

The OFB mode of operation does not extend ciphertext errors in the resultant plaintext output. One bit in error in the ciphertext causes only one bit to be in error in the deciphered plaintext. It is not self-synchronizing. If the two operations of encipherment and decipherment desynchronize, the system needs to be re-initialised. Such a loss of synchronization might be due either to the loss of correct boundaries of the j-bit blocks (because of a bit-slip) or an error in the value of variable X at one end or the other, causing the X values to differ at the two ends until re-initialisation takes place.

Each re-initialisation should use a value of SV different from the SV values used before with the same key.

This page intentionally left blank