ERROR CORRECTION FOR DIGITAL AUDIO RECORDINGS

PREPRINT NO 1991 (B4)

By Toshi T. DOI
Sony Corporation
Tokyo, JAPAN

Presented at the 73rd Convention 1983 March 15-18 Eindhoven, The Netherlands

AN AUDIO ENGINEERING SOCIETY PREPRINT

Error correction for Digital Audio Recordings

Toshi T. DOI

Sony Corporation

Tokyo, JAPAN

Presented at the 73rd Convention

1983 March 15-18, Einthoven,

Netherland

Abstract

Error Correction is one of the key technology in the field of digital audio. In this paper, the basis of error correction is explained plainly using "a supermarket shopping model", and various criterea are shown for the application of error correcting code to the digital audio systems. The details of code schemes are also explained in this article, such as the EIAJ format for recorders using video cassette, the DASH format for professional stationary head recorders,and the Compact Disc.

Cross Interleaving is an unique method of combining two codes by interleaving delay and is proven very efficient in performance as well as in hard ware design. Therefore, the method is applied to DASH format, the Compact Disc format and many other systems.

1. Introduction

One of the greatest features of digital audio technology is that the performance is decided by the number of quantization bit, and sampling rate, and is independent of the characteristics of the recording media, as long as word errors are corrected.

In other words, noise, distortion, and intermodulation which deteriorate the quality of analogue audio recording, are not directly related to the sound quality of digital audio recordings, they only make error rate worse. Even a system with terrible noise and distortion can make the highest performance recording if it is equipped with a strong error correction. However, if error correction is poor, and word error occurs at the rate of once every one million words for instance, the system cannot be used for sound recording because click is heard once every twenty seconds.

The history of digital audio, therefore, is supported by the progress of error correction technology. The error correction code itself has been well studied by the coding theorists, but its application to digital audio recordings is not so straight forward. So far, great effort has been paid to:

(i) the study of the criteria necessary for truely reliable systems in studio or in the home environment,
(ii) the investigation of the causes and the statistics of errors on magnetic tapes and discs,
(iii) the practical, the heuristic or the theoretical approach to the design of error correction code,
(iv) better trade-off between performance and hardware cost, and
(v) hardware design.

The purposes of this paper are

(1) to describe some of the essenses of the above progress,
(2) to describe the basis of error correction in a plain manner so that the average audio engineers without deep mathematical background can understand,
(3) to describe some of the applications of the error correction to the digital audio systems widely used in the world, namely the EIAJ-format for recorders using video cassette [1], the DASH-format for professional stationary head recorders [2] [3] and the Compact Disc (CD) [4].

The error correction codes may be categorized as follows:

(a) Linear code vs Nonlinear code.
(b) Block code vs Convolutional code.
(c) Word oriented code vs Bit oriented code.

In the application to digital audio recordings, the combination of "Linear-Block-Word oriented" methods has been mainly adopted. As an exception, the method called "Cross-interleave[5]" which uses a block code in a convolutional structure is applied to the DASH and the Compact Disc formats. The higher performance of convolutional structure is enjoyed by using simple structure of block code.

The reasons why word oriented code is used are as follows:

(i) Correctability for burst error is better.
(ii) Memory handling is simpler when RAM is used.
(iii) The digital audio system handles code word of 16 bit per sample, and hardware with 8-bit or 16-bit symbols is more simple for error correction.

In this paper, examples of error correcting code are shown by "a supermarket shopping model" for easier understanding of EIAJ, DASH and Compact Disc formats, omitting detailed mathematical discussion.

This is not a systematic approach, nor a theoretical explanation, and it is recommended that interested readers study appropriate writings [6] for deeper understandings of the codes presented here.

2. Code Error in Digital Audio Recordings

2.1 Causes of Code Error

The causes of code error in digital audio recording are attributed to the followings:

(1) Defects in recording media.
(2) Dusts, scratches and finger prints attached while in use.
(3) Fluctuation or irregurality in recording and reproducing machanisms.
(4) Fluctuation of the level of the reproduced signal.
(5) Jitter, wow and flutter.
(6) Noise.
(7) Inter-symbol interference.

Item (1) in the above list applies to various defects attached while producing tapes or discs.

In the case of magnetic tape, the following defects are found:

(a) Dusts and scratchings attached during production,
(b) Defects or scraping of magnetic materials including trace of dusts,
(c) Irregulality of tape edge or width,
(d) Trace of step between magnetic tape and leader tape.

In the optical disc, the followings are found:

(e) Defects in photo-resist.
(f) Dusts and scratches attached while cutting, developing, plating and pressing.
(g) Inappropriate strength of writing beam or developing time (asymmetry of pits).
(h) Forming error of pits at plating or pressing the disks.
(i) Bubbles, irregular refraction or other defects in the transparent disk body.
(j) Defects of the reflective metal coating.
(k) Irregurarity of back surface of the stamper or the mother.

Item (2) of the above refers to finger prints, dusts, or scratches attached while using the disc. In case of tape, the damage of tape edge is also included.

Item (3) means mis-tracking, mis-focusing (optical disks), un-locked servo, or fluctuation of the contact between the tape and head.

Items (4) and (5) of the above are mainly caused by item (3). It is also noted that small vibration of tape or head causes a great problem because the recorded wave length is very short.

The relation between error rate and white noise of item (6) is illustrated in Fig. 2.1.

The inter-symbol interference of item (7) is caused by the band width limitation and the non-linearity of the recording media.

2.2 Random Error and Burst Error

If a bit error does not have any correlation with other bit error, it is called a random bit error. When errors occur in a group of bits, it is a burst error.

When error is counted by word or block, random word (or block) error and burst (word or block) error are defined in a similar way.

Among the causes of error described in section 2.1, items (1), (2), (3), and (4) correspond to long burst error, and (5), (6), and (7), to random or short burst error.

In actual digital audio recordings, all kinds of error are mixed up, and the code should be designed to cope with a combination of random, short burst, and long burst errors.

2.3 Error Measurement

Measuring the meaningful error characteristics of digital audio recording media is not an easy task. In the early days, word errors were directly measured and optimization of error correcting scheme was carried out based on the obtained statistical data. The approach was theoretically reasonable but a real working system may not always represent the typical system. Moreover, the complete system may not be available in the process of system design.

Here, the most important point is to obtain the strength against accidents, such as tape damage, finger prints, scratches and so on. The system should be designed to obtain the block error rate better than, for instance, 10^{-4}, so as to guarantee to keep the error rate sufficiently low after the correction in the normal condition.

Defining the tolerable level of the accidents is the most important point in designing error correction schemes, which greatly depends on the structure of the recording media; cassette tapes, open reel tapes and optical disks must be handled completely differently in this sense. Fig. 2.2 shows one of the typical examples of optical disks measured under the following three conditions:

(1) Clean disk: at normal condition (with a block error rate of 3.3×10^{-4})

(2) Finger printed disk: finger print is attached all over the disc. (with a block error rate of 5.6×10^{-4})

(3) Scratched disk: the disk is rubbed on a wooden table for approximately one minute. (with a block error rate of 4.5×10^{-3})

In these cases of accident, long burst errors are observed, and therefore, it is recommended that errors be measured by block, rather than bit.

2.4 Statistical Model of Code Errors

Fig. 2.3 shows the well known statistical model for code error called the Gilbert Model [7]. It is a simple Markov chain, with basically two states, "Good (1)" and "Bad (2), (3)". There are two parameters, α and β, which represent the transition probability of falling into the state "Bad" and of recovering from it, respectively.

Even if the code is in the state of "Bad", there is a probability, "h" that the erroneous codes accidentally coincide with the correct ones. This probability is shown in Fig. 2.3 as State 2. It should be noted that, for bit error measurement, only state (3) is measurable and the burst length is often under estimated unless careful analysis is carried out.

This model can be applied to bit error, word error or block error, and in the case of bit error, $h \doteqdot 0.5$; otherwise $h \doteqdot 0$.

The following equations hold provided that the probability of error is sufficiently small.

$$\text{Error Rate} = \frac{\alpha(1-h)}{\alpha + \beta} \doteqdot \alpha(1-h) \qquad (2.1)$$

Code (Error) Correlation Coefficient

$$\rho \doteqdot 1 - \alpha - \beta \qquad (2.2)$$

Average burst length

$$L \doteqdot \frac{1}{1-\rho} \qquad (2.3)$$

It was found that the error distribution derived by this model did not accurately reproduce the measured error distribution of digital audio recordings, and thus a more precise model was proposed [8]. Nevertheless, this model has been used for most of the system evaluation because of its simplicity; furthermore, a precise model is not very meaningful when the system is mainly evaluated against accidents.

3. Error Detection

Parity check bit is well known as a simple error detection scheme. It detects 100% of even-number errors. This ability is not sufficient for most digital audio recordings, and therefore CRCC (Cyclic Redundancy Check Code) is commonly used for error detection.

Assuming that m-bit CRCC is attached to k-bit data, and a code word of n-bit (n=k+m) is formed (the redundancy is m/n), the error detectability is as follows:

(1) Burst error equal to or less than m-bit is always detected.
(2) Mis-detection probability of burst error of (m+1)-bit is 2^{-m+1}.
(3) Mis detection probability of burst error longer than (m+1)-bit is 2^{-m}.
(4) Mis-detection probability is 2^{-m}, when all error patterns occur equally.
(5) Random errors up to three bits is detected (depending on the selection of generating polynominals.)
(6) (1)~(4) of the above ability is not affected by the length of code word n ((5) is affected to a certain extent.)

How to decide the length of a block always poses a question. According to (6) of the above, the larger block seems better because it keeps the same detection capability

with less redundancy. But the system tends to encounter enough random or short burst errors frequently in some cases (like optical disks), then the resulting block error rate for longer block may worsen considerably. Moreover, even one bit error in a long block deteriorates the whole block.

There exists a trade-off between redundancy and the safety margin. The latter depends on the error correction strategy after detection.

CRCC is often used as "error pointer" (will be described later) of error correction code, and the detectability should be analized before designing the whole correcting schemes. Fig. 3.1~3.6 show the mis-detection probability of CRCC versus bit error correlation coefficient, where the bit error rate and the block length are taken as the parameters of error. Each curve is drawn with the CRCC length fixed.

4. Minimum Distance of Code

Fig. 4.1 shows an example of a three-bit code consisting of two data bits and one parity check bit. The code words form a set of four words out of eight possible combination of three bit words, and one-bit error always kicks the word out of the set.

The distance (Hamming distance) d between two words is defined as the number of positions where the two words have different bits. The code words shown in Fig. 4.1 have the distance of two between any pair of codewords (Fig. 4.2). This is called a code with minimum distance (d min) of two.

Generally, d min of a code with detetability of t_1-bit error satisfies the following equation:

$$d\ min \geq t_1 + 1 \qquad (4.1)$$

If the first bit of each word is missing, as shown in Fig. 4.3, the word is still recoverable by the remaining two bits. The missing bit is called the "erasure", and in this case a word with single erasure can be decoded correctly (single erasure correction). Generally, e-bit erasure correcting code must satisfy the following equation:

$$d\ min \geqq e+1 \qquad (4.2)$$

Fig. 4.4 shows that a code of five bits consists of two data bits and three check bits. In this case, a set of four words is selected out of 32 words so as to obtain a minimum distance of three (Fig. 4.5). Namely, the patterns derived by one-bit error from the coded words (shown in dotted line in Fig. 4.4) never overlap on each other. Therefore, one-bit error correction is easily performed by decoding a ROM containing error patterns.

Generally, a code with t_2-bit correction satisfies the following equation:

$$d\ min \geqq 2t_2+1 \qquad (4.3)$$

For the combination of t_1-bit detection, e-bit erasure correction, and t_2-bit error correction ($t_2<t_1$), d min should be as follow:

$$d\ min \geqq t_1+e+2t_2+1 \qquad (4.4)$$

The examples shown above are bit-oriented codes, which is exactly the same for word-oriented code; the distance d is defined between code blocks with d-word error, and the equations (4.1), (4.2), and (4.3) are defined for t_1-word error detection, e-word erasure correction, and t_2-word error correction, respectively.

If m words of check words are included in a block code, the minimum distance is limited to:

$$d\ min \leqq m+1 \qquad (4.5)$$

In the case of b-Adjacent code or Reed Solomon code (see Appendix), the equality in (4.5) holds.

$$d\ min = m+1$$

These codes are called the "maximum distance separable code"

5. Single Erasure Correction

Fig. 5.1 shows a typical shopping receipt where the prices of four goods (w_1, w_2, w_3, and w_4) and their total (P) are listed (SLIP (A).) The "Syndrome S" is calculated for checking. When there is no error, S=0.

In SLIP (B), the price of W_2 is missing (W_2*=0). The missing word is called "erasure". The erasure correction is very simple as shown in fig. 5.1.

SLIP (C) contains one word error in W_2, but the erroneous word cannot be specified from the receipt. The values of all words W_1, W_2,W_3, W_4, and P are uncertain, and therefore, they are expressed with an upper script ('). As the value of the Syndrome S is not zero, it is known that error occurred however, correction is impossible.

SLIP (D) includes the same error in W_2, but in this case the erroneous word is pointed out by some means (called error-pointer). The correction in this case, which is called pointer-erasure correction is exactly the same as that of erasure.

The code shown in Fig. 5.1 is a block code with four data words and one check word. The minimum distance is two (eq. 4.6), and the code is capable of one word error deteciton (eq. 4.1) and one erasure correction (eq. 4.2). However, error correction is impossible (eq. 4.3).

In actual codes used in digital audio recordings, a parity word of modulo-two addition (exclusive-or) is normally used for erasure correction, instead of the total value P in Fig. 5.1, and CRCC is used for error pointer.

This type of code has been applied to Sony's professional rotary head recorders [8], and 3M's digital mastering recorders [9].

6. Single Error and Double Erasure Correction

Fig. 6.1 shows the same kind of slip as Fig. 5.1 with two check words P and Q added. Since Q is a weighted total value, the single erroneous word can be easily found by comparing two syndromes S_1 and S_2. The correction is performed just as the erasure correction in Fig. 5.1.

Fig. 6.2 shows two word error with error pointers. The structure of the code is similar to that in Fig. 6.1 As is shown in the figure, double erasure is corrected by using two syndromes, S_1 and S_2, to solve two unknown values, E_2 and E_3 (error values on words W_2* and W_3*)

This code is a block code consisting of four data words and two check words, with a minimum distance of three (eq. 4.6). The code is capable of two word error detection (eq. 4.1), two word erasure correction (eq. 4.2), and one word error correction (eq. 4.3).

In the appendix, examples of b-Adjacent code and Reed Solomon code are shown, where the two check words are generated by a certain rule for each code.

The combination of double erasure correction code (Reed Solomon code with the minimum distance of three) and CRCC for error pointer has been adopted for Mitsubishi's stationary-head format [10], [11].

7. Convolutional Code

The examples shown in the previous sections 4.~6. are all classified into block code, where the encoding is completed within the block.

Fig. 7.1 shows an example of the encoder of convolutional code, where D means a delay of one word. The check words P_0, P_4, P_8,..... are generated once after every four words of data. Each check word is affected by the previous eight input words, as is shown below:

$$P_0 = W_0 + W_1 + W_2 + W_3 + W_{-4} + W_{-7} + W_{-10} + W_{-13} \quad .....(6.1)$$
$$P_4 = W_4 + W_5 + W_6 + W_7 + W_0 + W_{-3} + W_{-6} + W_{-9} \quad .....(6.2)$$
$$P_8 = W_8 + W_9 + W_{10} + W_{11} + W_4 + W_1 + W_{-2} + W_{-5} \quad .....(6.3)$$
$$P_{12} = W_{12} + W_{13} + W_{14} + W_{15} + W_8 + W_5 + W_2 + W_{-1} \quad .....(6.4)$$
$$P_{16} = W_{16} + W_{17} + W_{18} + W_{19} + W_{12} + W_9 + W_6 + W_3 \quad .....(6.5)$$

The decoder is shown in Fig. 7.2, and the syndromes corresponding to egs (6.1)~(6.5) are shown below:

$$S_0 = W_0 + W_1 + W_2 + W_3 + W_{-4} + W_{-7} + W_{-10} + W_{-13} - P_0 \quad .....(6.6)$$
$$S_4 = W_4 + W_5 + W_6 + W_7 + W_0 + W_{-3} + W_{-6} + W_{-9} - P_4 \quad .....(6.7)$$
$$S_8 = W_8 + W_9 + W_{10} + W_{11} + W_4 + W_1 + W_{-2} + W_{-5} - P_8 \quad .....(6.8)$$
$$S_{12} = W_{12} + W_{13} + W_{14} + W_{15} + W_8 + W_5 + W_2 + W_{-1} - P_{12} \quad .....(6.9)$$
$$S_{16} = W_{16} + W_{17} + W_{18} + W_{19} + W_{12} + W_9 + W_6 + W_3 - P_{16} \quad .....(6.10)$$

The upper script (') means that the values might be erroneous.

Supposing S_0 is the first non-zero syndrome after receiving a long enough period of data sequence, it may be concluded that errors occurred among W_0, W_1, W_2, and W_3. If there is a single error, a simple algorithm shown in Table 6.1 is used to find and correct the erroneous word.

Table 7.1 One Word Error Correction

Erroneous Word	Non-zero Syndrome	Correction
W_0'	S_0 and S_4	$W_0 = W_0' - S_4$
W_1'	S_0 and S_8	$W_1 = W_1' - S_8$
W_2'	S_0 and S_{12}	$W_2 = W_2' - S_{12}$
W_3'	S_0 and S_{16}	$W_3 = W_3' - S_{16}$

It is shown that the redundancy for this code is the same as that of the code in Fig. 5.1, but the error correction capability is superior. This is because the number of related syndromes for each error is bigger in the case of convolutional code due to the convolutional structure.

On the other hand, once uncorrectable error occurs, it would affect the syndromes long after the error passes away, and the number of resulting error in the decoded sequence becomes longer. This is called "Error Propagation".

In equation (6.1), the words used for forming the check word P are from W_0 to W_{-13}, and "the constraint length" of the code is defined as 14.

8. Interleave and Cross-Interleave

8.1 Interleave and De-Interleave

Interleave is a way to disperse the original sequence of bit or word into different sequence. The reverse process is called "de-interleave".

Fig. 8.1 shows an example of a simple delay interleave. The burst error occured during playback is converted into random errors by de-interleaving.

Interleave is often used with a block code to increase burst error correctability. Fig. 8.2 is a combination of a single erasure correction code and a delay interleave. All the words related to the original code block are dispersed to every sixth words, and burst error of up to six words can be corrected if appropriate error-pointer is provided.

A combination of double erasure correction code (b-Adjacent code with minimum distance of three), delay interleave, and CRCC for error pointer was adopted for EIAJ format [1], and is described in detail in Chapter 11 of this paper.

8.2 Cross Interleave Method

When two block codes are arranged two-dimensionally in the form where rows and columus form a big block, the code is called "a product code". The minimum distance of the final code is a product of that of each code.

The Cross Interleave Method falls into the class of product code, but distinguishes itself from the conventional one by the interleaved structure. The cross interleave method is a combination of two or more block codes which are separated from each other by delay for interleaving. In this case, the final correctability is sometimes better than conventional product codes owing to its convolutional structure.

Fig. 8.3 shows the general form of the Cross Interleave Method. At the decoder, the syndromes of one code can be used as the error-pointer for another code, and CRCC for error detection can be omitted in erasure correction. It is also possible to arrange the third block code after another delay interleave. This paper, however, focuses only on the combination of two codes.

In Compact Disc format, the two block codes are selected as Reed-Solomon code with minimum distance of five, and is named CIRC (Cross Interleave Reed-Solomon Code) [4]. This will be described in Chapter 13.

When both codes are selected as single erasure correction codes, the code is called "CIC (Cross Inrterleave Code) [5]". This method of Cross Interleave in general case (Fig. 8.3) has been developed from CIC.

8.3 CIC (Cross Interleave Code) [5]

Fig. 8.4 shows a simple example of the encoder of CIC, which is similar to Fig. 8.2, but another single erasure correcting code is added after the delay for interleaving.

Thus the check words P and Q are generated by the following equations.

$$P_0 = W_0 + W_1 + W_2 + W_3 \quad \text{.....(8.1)}$$
$$P_4 = W_4 + W_5 + W_6 + W_7 \quad \text{.....(8.2)}$$
$$P_8 = W_8 + W_9 + W_{10} + W_{11} \quad \text{.....(8.3)}$$

$$Q_0 = W_0 + W_{-3} + W_{-6} + W_{-9} + P_{-16} \quad \text{.....(8.4)}$$
$$Q_4 = W_4 + W_1 + W_{-2} + W_{-5} + P_{-12} \quad \text{.....(8.5)}$$
$$Q_8 = W_8 + W_5 + W_2 + W_{-1} + P_{-8} \quad \text{.....(8.6)}$$

It is easier to understand CIC by a code diagram shown in Fig. 8.5, where the solid lines represent the word sequence which forms the check words P, and the dotted lines, Q.

Supposing W_0 and W_{-3} are erroneous, collection only by Q-sequence is impossible. The syndrome, however, is non-zero as long as the error patterns of W_0 and W_{-3} do not coincide and is used as error-pointer for decoding P_{-4} and P_0 sequences. Since both codes are single erasure collecting code, the error is efficiently corrected.

The correction is impossible, in the worst case, by four-word error, for instance, W_{-3}, W_{-2}, W_0, and W_1. In this case, for all related sequences, P_{-4}, P_0, Q_0, and Q_4, double word error is found. In other words, this code is capable of correcting arbitrary triple-word error.

The correctability of CIC depends on the steps of decoding, one step decoding is as good as single erasure correction. The correction capability increases as the number of steps increases. Fig. 8.6 shows one example of the capability plotted versus number of decoding steps.

Fig. 8.7 shows another example of the correctability of CIC against burst error. In this case an additional delay interleave is connected after the encoder of CIC. The delay values for each line are D, 2D, 3D, ..., and 7D (D=16 or 32), while the unit delay of the interleave inside the CIC encoder is assumed to be "d" (d=2 or 6). .....see Fig. 12.1.

Professional stationary head format "DASH" proposed by Sony [2], [3] and AHD (Audio High Density) disk system [12] by JVC have adopted this code.

8.4 ICIC (Improved Cross Interleave Code) [5]

The ability of CIC can be improved by the following two modification as shown in Fig. 8.8. This modified version of CIC is called ICIC (Improved Cross Interleave Code).

(1) The amount of delay for interleave is increased by D in CIC (Fig. 8.4), while it is irregular in ICIC (Fig. 8.8).

(2) In forming the check word P , only the data words are used in CIC. In ICIC words Q are added.

Fig. 8.9 shows a code diagram of ICIC, and the worst case of uncorrectable error consist of six words forming a loop such as Q_0, W_0, W_{-18}, W_{-19}, Q_{-4}, and W_{-4}. An arbitrary error consisting of up to five words can be corrected in ICIC.

The irreguraly of delay shown in Fig. 8.8 is very important, and thus the loop of un-correctable error should not be made shorter than six. The principle is as follows:

(a) For preparation, a series of the differences between adjacent delay value is made. in the case shown in Fig. 8.6, the series is

$$a_1=1,\ a_2=3,\ a_3=2,\ a_4=6,\ a_5=7 \qquad \ldots\ldots(8.7)$$

These number corresponds to the jump of dotted line in Fig. 8.9.

(b) Minor loops shorter than six words do not occur in the code diagram shown in Fig. 8.9 on the condition that any three continuous values in (8.7) do not satisfy eq. (8.8):

$$\left.\begin{aligned} a_i+a_{i+1}&=a_{i+2} \\ a_i&=a_{i+1}+a_{i+2} \end{aligned}\right\} \quad (8.8)$$

Namely;

$$\left.\begin{aligned} &a_1+a_2\neq a_3 \\ &a_1\neq a_2+a_3\neq a_4 \\ &a_2\neq a_3+a_4\neq a_5 \\ &a_3\neq a_4+a_5 \end{aligned}\right\} \quad (8.9)$$

The original two codes, P and Q, are single erasure correction code described in Chapter 5 with minimum distance of two. Therefore, if a product code is formed by these two codes, the resulting minimum distance should be four (2X2). The code is then capable of correcting a single word error or triple word erasures. In the case of ICIC, it is remarkable the code is capable of correcting quintuplet word error.

Fig. 8.10 shows one of the examples of the correctability of ICIC (Fig. 8.6); better performance is obtained after four steps of decoding. In other words, if decoding step are limitted to less than three steps, significant difference are not be found between CIC and ICIC.

Fig. 8.11 shows the ability of ICIC against burst error, while the additional interleave is similar to that in Fig. 8.7.

8.5 Block Completed Cross Interleave

CIC, ICIC or any kind of Cross Interleave Code can be easily modified into a block structure. Fig. 8.12 shows an example of block completed CIC, modified from the code described in Figs. 8.4 and 8.5.

The correctability against random error is the same as that of the original CIC. Correctability against burst error can also be kept the same if the block length is long enough. The memory necessary for encoding and decoding is doubled in this structure, however. If this large block code is compared with the optimum block code with the same size and redundancy, the performance is no doubt weaker, but for the practical application it has a merit that the encoder and decoder are much simpler.

This kind of structure is adopted to the digital sound track of 8mm-video which is recorded on video track in time-compressed manner; the block structure is required by its time-compression method [13].

The professional rotary head format of JVC [14] has also adopted this structure.

9. Error Concealment [15]

When error exceeds the ability of error correction code, the un-corrected words should be concealed.

Fig. 9.1 shows the noise power induced by various interpolation method applied to pure tone signal with uncorrectable errors.

The methods of concealment used in Fig. 9.1 are as follows.

(1) Muting. The value of erroneous word is always zero.

(2) Previous value holding or zero'th order interpolation.

(3) Mean value interpolation or first order interpolation. To replace the erroneous word by the mean value of the previous and the next words.

(4) N'th order interpolation. To use an polynomial of n'th order instead of the first order in (2).

In most systems now in use, the combination of (2) and (3) is used. If all the erroneous words are neighbored by words without error (see Fig. 8.1), (3) is applied. For the consecutive error words, (2) is applied with the final erroneous word interpolated by (3).

It is also well known that by using an appropriate digital filter, a better interpolation is possible.

Fig. 9.2 shows the result of a subjective test [15]. It is shown that the length of error does not greatly affect the perception as long as all interpolated words are neighbored by words without error.

10. Evaluation of Error Correcting Schemes

The fundamentals of error correction are described in the previous chapters. In the application to actual systems, these methods are mixed, and the scheme is optimized to suit the particular system either by computer simulation or by trials and errors. The most important point in designing the coding scheme is, therefore to set the criteria for the performance of the code. Some of the criteria are shown below.

(A) Probability of mis-detection

(B) Maximum burst error to be corrected.

(C) Maximum burst error to be concealed.

(D) Correctability of random error.

(E) Correctability for the mixture of random, short burst and long burst error.

(F) Guard space.

(G) Error propagation.

(H) Block size or constraint length.

(I) Redundancy.

(J) Ability for editing.

(K) Delay for encoding and decoding.

(L) Cost and complexity of the encoder and decoder.

The details of each item are described below.
For most digital audio systems, errors exceeding the correctability of the code are designed to be concealed. Therefore, the most important item is the probability of mis-detection (A). The mis-detected error cannot be concealed and results in an unpleasant click noise from the speakers. Mis-detection should not occur even in the worst case of accident.

Items (B) and (C) represent the strength against burst errors.

Table 10.1 gives some examples of the burst error correction and concealment for EIAJ [1], DASH [2], [3], and CD [4] format. In each case, the length of concealment is designed considerably longer than that of the correction. It is shown that the length of DASH is longer than others, because EIAJ is designed for cassette tape where the chance of attaching finger prints is quite low. CD is strong against finger print because the thick transparent layer (1.2mm) between signal pits and the surface nullifies the effect of finger prints.
The format of DASH, on the other hand, is for open reel recorders with the capability of splice editing; the damage caused by finger prints is severe.

The burst error correcting capability is mainly determined by delays if interleaving is used with the correcting code. The correction or concealment length should be evaluated as a function of the memory size required.

Fig. 10.1 shows the correctability of various codes against random block error (D) where one block consists of 288 bits. The block length is supposed to be long enough so that even short burst error could be treated as random block error. Initially the evaluation shown in Fig. 10.1 is considered as appropriate, but it was found not to coincide with the experience in studio environment. The reasons are shown in the followings.

If block error rate is measured for the recorded wave length from 1.6 to 2.0 μm, the average value may be 10^{-5}, which, referring to Fig. 10.1, means that mis-correction occurs once every five hours for single-erasure correction, once every sixty years for double-erasure correction, and once every two million years for Cross Interleave Code.

Machines installed in studios, however, show much frequent mis-correction rate than these values. The reasons for this are:

(1) the tapes are not always new,
(2) dusts and finger prints are more severe in studio environment,
(3) the machines are not always tuned properly.

Therefore, the evaluation for the mixture of random, short burst and long burst (E) proves to be much more realistic [2], [3], [17]. The Gilbert model shown in Fig. 2.3 helps this evaluation performed on the plane with two axes of error rate and error correlation.

Format	Type	Correction	Concealment		Reference
			Good	Marginal	
EIAJ	Tape, Rotary Head	4,096 bit	-	8,192 bit	[1]
DASH	Tape, Stationary Head	8,640 bit	33,982 bit	83,232 bit	[2],[3]
CD	Disc, Optical	3,874 bit	13,282 bit	15,495 bit	[4]

Table 10.1 Burst Error Correction and Concealment

Fig. 10.2 shows one of these planes with two axes of block error rate and block error correlation coefficient. The area A is the measured value of new tapes, and the area B shows tape wearing and mis adjustment of machines is also considered to be located in the area B. On the same plane, the correctability of DASH format is shown [2], [3]. The region (i) is the area with the block error rate better than 10^{-8} and the deterioration of sound quality is not expected even with some concealed error words. The region (iii) is the area with the error rate after crrection worse than 10^{-4}, and the recording is impossible.

In between the regions (i) and (iii), there is a region of warning (ii) where the recording can be done. However, it should be carefully checked because in this region, either the tape or the machine may be in bad condition.

Guard space (F) is defined as the area without any error which is needed to perform a certain error correction. If guard space is larger, the possibility of another error happening there becomes greater, and consequently, the correctability gets poorer at high error rate. The results of computer simulation usually include this probability of guard space error as shown is Fig. 10.2. Fig. 10.3 shows the actual area of guard space on tape for the burst correction and concealment of DASH format [2]. For perfect correction, the guard space is required to be approximately three times of the correctable area. Those for good and marginal concealment are shown to be much smaller, namely, two times and one eights of concealment area, respectively. This shows that the strength against accidents is kept by several levels of protection means.

Error propagation (G) is a problem only for convolutional codes. The CIC, ICIC, CIRC, or any other structure of the Cross Interleave however, enjoy the high performance of convolutional code but free from error propagation. This is because of their block-oriented structure, namely the uncorrectable error for one sequence of the block code always turns out to be one-word error to the next sequence, and even the simplest CIC can correct it.

Items (H), (I) and (L) are inter-related to each other. Fig. 10.4 shows the relation between correctability and redundancy for single and double erasure correction code. The check words required for double erasure correction are two, and that for single erasure is one (Eg. 4.6).

By comparing two codes at the same redundancy, it is clear that the correctability of double erasure correction is much better (Fig. 10.4). The size of a block is doubled in double erasure correction code.

In other words, with the same redundancy, the more complex the code with bigger block size the stronger its correctability. Therefore, it is a trade off between the performance, the redundancy and the cost of the encoder and decoder.

The ability for editing (J) should be considered in designing error correction schemes. If inexpensive editing is to be performed, convolutional structure is not recommended.

If tape splice editing is required, some special means should be designed for error correcting schemes because there is a certaion length of area along the spliced point where the signal cannot be reproduced. Fig. 10.5 shows the method applied to DASH format. Odd and even number of words are displaced by a large delay at interleaving. After de-interleaving, the lost area is always backed up by each other and there is an overlapped area of both information in order to perform cross-fading at the spliced point [2]. The protection against finger print should be also well considered because lots of finger print are expected around the spliced point.

Punching in/out is another important point to be considered for professional multi-channel recorder design. Fig. 10.6 shows typical procedure of punching in/out in digital recorders. Namely, the signal read from playback head is re-recorded at approximately the same position on the tape. At the entry point of this recording, one block may be destroyed by the in-accuracy of the distance between heads or by jitter. If this is not considered many blocks will be destroyed after punching in many times at the same point, causing some trouble.

In DASH format, this entry point is designed to coincide with one of the series of the Cross Interleave Code; the ability of single erasure correction is always kept for the destroyed blocks.

Delay for encoding and decoding is important in considering the rise up time of recorders; delay time of 0.1~0.2 seconds are tolerated.

11. Error Correction on the EIAJ format

EIAJ-format is designed for home-use digital audio processor to be connected to video cassett recorders. Horizontal and vertical pulses are added to digitally coded signal,and pseudo-video signal is formed to be fed into video cassette recorder [4].

Fig. 11.1 shows the encoder of the format. The scheme is basically a combination of a double erasure correction code (b-adjacent code with minimum distance of three), a delay interleave, and CRCC for error pointer.

Two check words, P and Q, are formed for six data words. P is summation (modulo-two) of all data words, and Q is weighted summation (Fig. 6.1) formed by a matrix circuit shown in Fig. 11.2. The details of b-adjacent code are described in the Appendix.

After going through delays for interleaving, CRCC is encoded with these eight words. The block after the delay interleave forms a data stream on each horizontal line.

Fig. 11.3 shows the data sequence on tape, where Ln and Rn are words for left and right channels. Hn represents each block on the horizontal line. In between these blocks, another fifteen blocks exists but are not shown in this figure. The words related to a coded block are recorded after every sixteen blocks.

Since double erasure correction code is capable of correcting two words in the block before delay, the code is capable of correcting burst error of 32 blocks.

Fig. 11.4 shows the performance of EIAJ format, where the bit error rate is fixed to 10^{-4}, and bit error correlation coefficient is chosen as the parameter. Single erasure correction does not use the weighted sum Q. One data word of the EIAJ format consists of 14 bits. However, word Q can be used to record two extra bits from each word and thus 16 bit can be recorded, provided that the reduction in correctability shown in Fig. 11.4 is tolerable.

Cross word correction [15], [18], [19] is a method to decode the code comparing the patterns of two syndromes. In this case, CRCC is not only used for error pointer but also for error correction. As is shown, considerable improvement in the performance is obtained comparing to double erasure correction. However, it was not adopted to the realized machines because of its complexity of the decoder circuit.

12. Error Correction on the DASH format

DASH (Digital Audio Stationary Head) is a format for professional open reel tape recorders covering a wide range of versions; the channel number from 2 to 48, and the tape speed, from 12cm/s to 76cm/s [2],[3].

The major point to be considered in designing this format is the editing capability (Chapter 10, J). The signal processing of punching in/out is to be done as shown in Fig. 10.6, and therefore the total delay of encoding and decoding defines the distance between the playback and recording heads. From the point view of mechanical design, this distance cannot be too long, and consequently the total delay is limited to within one tenth of a second.

Fig. 12.1 shows an example of CIC (Cross Interleave Code) encoder combined with CRCC and another delay interleaver. The CIC is coded to k-word of data ($W_0 \sim W_{k-1}$), and two check words P and Q are formed by the exclusive-or circuit. The unit delay for the interleave inside CIC is "d", and the outside, "D".

Fig. 12.2 shows an encoder and a decoder of double erasure correction code with combination of CRCC and a delay interleaver, which is identical with the EIAJ format (Fig. 11.1).

Some parameters of these codes are given in Table 12.1. CIC provides better capability for random-word error, while the burst error correctability is half the double erasure correction code under the condition that the total delay is restricted to a certain value.

Figs. 12.3~12.7 show the comparison between CIC (Fig. 12.1) and double erasure correction code (Fig. 12.2) for the different values of k, D, and d. Here the decoder of CIC is assumed as double steps.

The cross point of the curves of the two codes always exist between the bit error correlation coefficient ρ of 0.99 and 0.999. For the same value of k, the point shifts to a larger value of ρ as the value of D increases. For instance, at the parameters of k=6, D=14, d=4, the cross point is ρ=0.997 (fig. 12.5), which means CIC has the better performance when average burst length is below 300 bits (Eg. 2.3).

With these comparison, the CIC has been adopted instead of the double erasure correction code, because long burst error over 300 bits is observed by finger prints or head crogging, and even the capability of double erasure correction is not sufficient. Furthermore, concealment method which is practical against accident which seldom happen should be

	Total Delay of Encoder/Decoder (Words)	Burst Error Correction Length (Blocks)	Random Word Error Correction (Words)	Figure
CIC with Interleave	(k+1) (D-d)	D	3	12,1
Double Erasure Correction with Interleave	(k+1) D	2D	2	12,2

Table 12,1 Total Delay and Error Correction of Codes (Fig. 12,1 and 12,2) provided for very long burst.

Burst error less than 300 bits are more frequently found by various causes described in the Chapter 2.

ICIC was given up by the following reasons:

(1) For an economical design of machines the steps of decorder are limited, for instance, to three, and the significant difference cannot be found between CIC and ICIC (Fig. 8.6, 8.10).

(2) At the edited point of punching in/out or splicing, the decoding of ICIC is much complicated and requires more circuitry.

In figures 12.3~12.7, it is clear that the performance of CIC is much better when k=4 than k=6. This is due to the following reasons:

(i) Under the limitation of the total delay (Table 12.1), D can be larger for smaller k.

(ii) Block error rate is better in small block size for the same bit error rate at smaller bit error correlation coefficient.

The number of k, therefore, is a trade-off between the performance and the wave length to be recorded. Fig. 12.8 shows the relationship between the number of "k" and the wave length, where one synchronization word and one CRCC are added to two of the coded blocks in each case. The density ratio of the channel coding is assumed as 1.5 [20].

A typical example of error rate vs. packing density is also shown in Fig. 12.8, and by comparing these, k is chosen to be six.

Fig. 12.9~12.26 show some of the results of the computer simulation for Cross Interleave Code with various parameters of D and d, and the steps of decoding. In order to evaluate the performance for long burst error, block error correlation coefficient is taken for the evaluation parameter.

It should be noted that in some combination of D and d, the correctability becomes suddenly worse (Fig. 12.10 and 12.14). This is because the two sequences of P and Q in CIC cross the same block, and thus the full ability of the three word correction cannot be obtained.

According to the results of these simulation the parameters D and d are chosen to be 17 and 2, respectively.

In order to perform tape-splice editing, another consideration is required to recover the lost area of splicing, as well as to make cross fading at the edited point. A large delay of "$\mathcal{D}$" is inserted in DASH format between the even and odd number of words to solve these problems (Fig. 10.5).

Figures 12.27 and 12.28 show an example of a machine design; the recording and playback systems. Here the detection of the splice point and the cross fading are shown.

The encoding scheme of DASH with delays of d, D, and $\mathcal{D}$ is shown in Fig. 12.29, where the delay length is represented allegorically by length of arrow.

Two check words P and Q are defined as follows.

$$P(x)=W(x)+W(x+4)+W(x+8)+W(x+2)+W(x+6)+W(x+10) \quad \ldots(12.1)$$

$$Q(x)=W(x+4d\cdot k)+W(x+4+3d\cdot k)+W(x+8+2d\cdot k)+P(x+d\cdot k)+W(x+2-d\cdot k)+W(x+6-2d\cdot k)+W(x+10-3d\cdot k) \quad \ldots(12.2)$$

Where k=12, and x=12n+1; or x=12n+2, n=0,1,2...

Fig. 12.30 shows the structure of one block on the tape, and the sequence of the synchronization word, data words, check words and CRCC.

The format DASH consists of several versions with word distribution of one, two, or four tracks per channel. In each case the error correcting scheme for one track is exactly the same as above. In other words, even if the data words of one channel is recorded on two or four tracks, the error correction scheme is formed independently of each track; and even if the error on one track exceeds the correctability of the code, the ability of the other track is not affected.

13. Error Correction on the CD (Compact Disc) format

Among the causes of error in the optical disks, the effect caused by finger prints, scratch, or dusts are not so different from that in magnetic tapes. Therefore it is sufficient if the error correction scheme enables correction of considerably long burst error with remarkably small guard space.

On the other hand, the causes described in (e)~(k) in Chapter 2 have the tendency to produce random or short burst error. In addition to that, there is a possibility of producing random or short-burst error by mis-focusing and tracking offset, because they might deteriorate the frequency characteristics and signal to noise ratio.

The study started from the comparison between the ICIC and a newly developed convolutional code [21]. ICIC has the ability of correcting arbitrary five word error (Chapter 8), while the convolutional code can correct up to two symbols (words). Both codes are designed with the same redundancy to obtain a bit rate of 2.35Mb/s. Both performed very well for normal discs with average finger prints and scratches. It is found that the burst error correction is better with the ICIC, because the correctable length is three time longer than the convolutional code, and if concealment is considered, it becomes more than seven times (Fig. 13.1).

In order to improve the yields in disc production, the strength against random or short burst error should be as good as possible, but the ICIC has some limitation. This limitation comes from the pointer erasure structure which the block error rate becomes extremely high for random error with high rate.

Fig. 13.2 shows the relationship between bit error rate and block error rate for various size of block. This type of block configuration will show terrible results if one block is larger than 100 bits and random bit error is worse than 10^{-3}, for instance. The situation will be improved if the size of the block is made smaller, but this is impossible from the limitation of the redundancy.

The above situation was solved by the author, by expanding the concept of cross interleaving. A new code named CBAC(Cross B-Adjacent Code), which is a combination of a couple of double erasure (single error) correction code with delay interleave in between, was proposed [22].

Random or short burst error can be corrected in a block without using CRCC pointer, and absurdity shown in Fig. 13.2 is completely solved.

The bit rate can be reduced because CRCC is omitted and one word (16 bits) of data is devided into two simbols (8-bits) for error correction.

The number of symbols in one block, "k", is also an important factor with this code. It is shown in Fig. 13.1 that for larger number of k, the redundancy and bit rate become lower; at the same time, the efficiency of correcting burst error deteriorates.

At the final version, k is chosen to be 24, and the code is expanded into a quadruple erasure (double error) correction code (Reed Solomon code with the minimum distance of five), which is named CIRC (Cross Interleave Reed-Solomon Code) [4], [23], [24], [25].

The efficiency of burst error correction and the bit rate of CIRC is plotted on Fig. 13.1.

Fig. 13.3 shows the encoder of CIRC, the details of which are discribed below.

(1) 12 data words are shown in L and R which represent left and right channel of stereo sound.
(2) Each data word is devided into two symbols, which are indicated by W.
(3) A delay of two symbols are given in order to separate even and odd data words. This is effective when error is uncorrectable and concealment should be taken.
(4) The order of the 24 symbols is re-arranged so as to separate even and odd data words as far away from each other as possible.
(5) Four check symbols Q_{12n}, Q_{12n+1}, Q_{12n+2}, and Q_{12n+3} are formed in the C2-encoder; the Reed-Solomon code, with the minimum distance of five.
(6) Delay interleave is performed, where the unit delay D is four.
(7) Check symbols of P_{12n}, P_{12n+1}, P_{12n+2}, and P_{12n+3} are formed in the C1-encoder.
(8) A delay of one symbol is inserted to every other line. This is effective to separate the two adjacent symbol error, which may occur by a short burst at the boundary of frames, into different block coded by the C1-encoder.
(9) The eight check symbols are inverted. This is effective to prevent producing all-zero frame for all-zero data words.

The check symbols in the C2 and C1 encoders are generated to satisfy the following equation.

$$H \cdot V = 0 \qquad (13.1)$$

Where H is called the parity check matrix and V is a vector including data and check symbols. The values of H for C2 (=Hq) and that for C1 (Hp) are shown in Fig. 13.4, while the vectors V for C2 (=Vq) and C1 (=Vp) are shown in Fig. 13.5.

The decoder of CIRC is shown in fig. 13.6 which is the opposite signal processing of the encoder.

The performance of CIRC depends on the strategy as well as on the steps of decoding. Fig. 13.7 shows one of the examples of the correctability and the detectability of CIRC against random symbol error, here the decoding named "Super Strategy" [25] is adopted.

Table 13.1 shows the details of this strategy. It is important to note that although full ability of the code in C-1 decorder is utilized (double error correction), error pointers are set in order to avoid mis-correction, and the results are checked in C2-decoder by comparing these pointers with the location found by decoding.

This strategy has proven very effective both for detection and correction of errors.

Table 13.1 Super Strategy for CIRC Decoding

```
     f:  number of error pointer input into the C2 decoder
     Lc: number of error pointer agreeing with the error
         locations calulated from the syndrome
C1 decoder
     if single-or zero-error-syndrome
     then modify at most one symbol accordingly
     else if double-error-syndrome
          then modify 2 symbols accordingly
               assign error pointer to all symbols of
               the received word.
          else assign error pointer to all symbols of
               the received word.
C2 decoder
     if single- or zero-error-syndrome
     then modify at most one symbol accordingly
     else if  f≤4
          then if double-error-syndrome and Lc=2
               then modify 2 symbols accordingly
               else if {double-error-syndrome and
                       [(Lc=1 and f≤3) or (Lc=0 and f<2)]}
                       or (f≤2 and not double-error-syndrome)
                    then assign error pointer to all symbols
                         of the received word.
                    else copy C2 error pointer from C1 error
                         pointer.
          else copy C2 error pointer from C1
```

The ability of CIRC for long burst is already shown in Table 10.1. It is easily understood that the correctability is 4D blocks (D=unit delay in Fig. 13.3), since the code is the guadruple erasure correction code.

The ability to conceal a long burst is, on the other hand, decided by the distance between even and odd words on the disc (Fig. 13.3), which is also shown in Table 10.1.

14. Conclusion

The fundamentals of error correction and its application to the digital audio recording systems are described in this paper. The stress of the explanation is laid on the practical side omitting detailed theorelical consideration.

However, according to the author's experience, the optimum code in theory is not always optimum in practice, and, a much wider view is required in the practical applications. In this article, therefore, the basis of code theory is explained by a "super market shopping model" and much attention is paid to the explanation of how to evaluate the code.

The Method of "Cross Interleaving" was proposed by the author and his team, and is now widely used in the field of digital audio, because of its simplicity and efficiency.

Three code formats widely used in the world, namely EIAJ, DASH and CD, are explained in detail with some side information. The other formats applied to the systems in the market are briefly mentioned with explanation ralating to those codes.

Looking back to the short history of error correction code applied to digial audio recordings, a remarkable progress has been made with high enough performance and relatively low redundancy. The further progress will be necessary when the packing density increases and heavy erroneous recording systems are introduced.

One of these applications will be domestic cassette recorder which is expected to become competitive to conventional analogue cassette in size and playing time, while obtaining high performance equivalent to CD.

Acknowledgement

The author wishes to thank Messeurs K. Odaka, S. Furukawa, M. Tanaka, G. Fukuda of Sony Corporation and L. B. Vries of N. V. Philips for their contribution to the development of various codes as well as to the evaluation method described in this paper.

References

[1] G. Fukuda, K. Odaka, and T. T. Doi; "On Error Correctability of EIAJ-Format of Home Use Digital Tape Recorders", 64th AES Conv., No. 1560 (G-5), Nov. 1979

[2] T. T. Doi, Y. Tsuchiya, M. Tanaka, and N. Watanabe; "A Format of Stationary-Head Digital Audio Recorder Covering Wide Range of Application," 67th AES Conv., No. 1677 (H-6), Nov. 1980

[3] T. T. Doi; "A Design of Professional Digital Audio Recorder", 71th AES Conv., No. 1885 (G-2), March 1982

[4] L. B. Vries, K. A. Immink, J. G. Nijboer, H. Hoeve, T. T. Doi, H. Ogawa, K. Odaka; "The Compact Disc Digital Audio System: Modulation and Error Correction," 67th AES Conv., No. 1674 (H-8), Oct. 1980.

[5] T. T,. Doi, K. Odaka, G. Fukuda, S. Furukawa; "Cross Interleave code for Error Correction of Digital Audio Systems", 64th AES Conv. ,No. 1559 (H-4), Nov. 1979.

[6] For Instance:
1 S. Lin; "An Introduction to Error-Correcting Codes", Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1970
2 E. R. Berlekamp; "Algebic Coding Theory", McGraw-Hill Book Co., New York, 1968

[7] E. N. Gilbert; "Capacity of a Burst-Noise Channel", BSTJ, 39, 5, PP1253~1263, Sep. 1960
[8] T. T. Doi, Y. Tsuchiya, and A. Iga; "On Several Standards for Converting PCM Signals into Video Signals", JAES, 26, 9, Sep. 1978
[9] J. A. McCracken; "A High Performance Digital Audio Recorder", AES 78th Conv., No. 1268 (N3), Nov. 1977
[10] K. Tanaka, T. Yamaguchi, Y. Sugiyama; "Improved Two Channel PCM Tape Recorder for Professional Use", AES 64th Conv., No. 1533 (G-3), Nov. 1979
[11] K. Tanaka, Y. Kusunoki, Y. Sugiyama, O. Nakajima, T. Furukawa, and S. Kunii; "On PCM Multi-Channel Tape Recorder Using Powerful Code Format", AES 67th Conv., No. 1690 (H-5), Nov. 1980
[12] N. Takahashi; "Digital Audio Disc System (AHD) Emploies an Electro tracking Capacitance Pickup", IECEJ, Technical Report, EA-81-26, July, 1981.
[13] K. Nakano, H. Moriwaki, T. Takahashi, K. Akagiri and M. Morio; "A New 8-Bit PCM Audio Recording Technique Using an Extension of the Video track", IEEE International Conf. on C. E., June, 1982
[14] Y. Yamada, Y. Fujii, M. Moriyama, and S. Saitoh; "Professional-Use PCM Audio Processor with a High Efficiency Error Correction System", AES 66th Conv., No. 1628 (G-7), May 1983.
[15] G. Fukuda, T. T. Doi; "On Drop out Compensation of PCM System...Computer Simulation method and a New Error Correcting Code (Cross Word Code)", AES 60th Conv., No. 1354 (E-7), May 1978
[16] T. Sonoda, and Y. Tsuchiya; "A Listening Test of Drop out Compensation of PCM Audio Recorder", National Conf. of ASJ, No. 3-2-13, Oct. 1977
[17] T. T. Doi, Y. Tsuchiya, and G. Fukuda; "Statistical Analysis of Error Correcting Schemes of Stationary Head Digital Audio Recorders", Joint Conf. of ASA and ASJ, No. MM13, Nov. 1978
[18] T. T. Doi; "Crossword Decoding of Cross-Interleave Code", ASJ Conf., No. 3-5-13, June, 1979
[19] T. T. Doi; "Crossword Decoding of Interleaved b-Adjacent Code", ASJ Conf. No. 3-5-14, June 1979
[20] T. T. Doi; "Channel Codings for Digital Audio Recordings", AES 70th Conv., No. 1956 (I-1), Nov. 1981
[21] L. B. Vries; "The Error Control Systems of Philips Compact Disc", AES 64th Conv. Nol 1548 (G-8), Nov. 1979
[22] T. T. Doi; "Cross B-Adjacent Code", internal paper for a joint meeting of Sony and Philips, Jan. 1980

[23] L. B. Vries and K. Odaka;
"CIRC, The Error Correcting Code for the Compact Disc Digital Audio System", presented at AES Conference in Rye Town, June 1982

[24] T. T. Doi;
"The Signal format of Compact Disc digital Audio", IECEJ Technical Report EA 81-23, July 1981

[25] K. Odaka, T. furuya, and A. Taki;
"LSI's for Digital signal Processing to be Used in Compact Disc Digital Audio Players", AES 71st Conv., No. 1860 (G-5), March 1982

[26] T. T. Doi, A. Iga;
"Digital Audio", Radio Technology Inc., April 1982

APPENDIX A B-adjacent Code

B-Adjacent code is designed to correct adjacent b-bit word error, and is known as maximum distance separable code (Chapter 4, eq. 4.6).

Fig. A1 shows an example of one word error and double erasure correction code with using two check words P and Q (see Fig. 6.1 and 6.2). Where the check word P is a simple summation (+: modulo-two sum) and Q is the weighted summation. Instead of using the weighting factors 4, 3, 2, and 1 in Fig. 6.1, b-Adjacent code uses those of the multiples of matrix T, namely T , T , T , and T in Fig. A1.

The matrix T and the execution circuit of TA (A: vector) is shown in Fig. A1, and T^n is defined as an execution of the circuit n-times.

The syndromes are as follows, where the upper script (') means suspicion of error.

$$S1 = A'+B'+C'+D'+P' \quad .....(A1)$$

$$S2 = T^4A'+T^3B'+T^2C'+TD'+Q' \quad .....(A2)$$

Hence the location of single word error is detected by checking the following formulas.

Table A1 Single Word Correction

Error Location	Syndrome
A'	$T^4 S1=S2$
B'	$T^3 S1=S2$
C'	$T^2 S1=S2$
D'	$TS1=S2$
P'	$S1\neq 0,\ S2=0$
Q'	$S1=0,\ S2\neq 0$

The procedure for double erasure correction is also similar to the "supermarket shopping model" shown in Fig. 6.2. Supposing B' and C' are pointed out as erroneous, and their error patterns are E_b and E_c, the correction is executed by the following equations.

$$S1 = Eb+Ec \quad .....(A3)$$

$$S2 = T^3Eb+T^2Ec \quad .....(A4)$$

$$\therefore \quad Eb = (T^3+T^2)^{-1}(T^2S1+S\ ) \quad .....(A5)$$

$$Ec = S1+Eb \quad .....(A6)$$

Usually the values of inverted matrixes such as $(T^3+T^2)^{-1}$ are calculated beforehand for the all possible combination and are stored in ROM.

In case of "super market shopping model" on Fig. 7.1 and 7.2, the number of words within one block in limited to 9 if numbers 1~9 are used for weighting factors.

The maximum number of words in a block for b-Adjacent code is decided by the selection of generator polynomials.

In this example (Fig. A1), that is,

$$G(x) = x^7+x^3 \qquad \text{.....(A7)}$$

and the block should be less than 12 words, because

$$T^{13} = T. \qquad \text{.....(A8)}$$

If appropriate polynomial (irreducible-) is selected, the maximum block size for b-bit word is

$$2^b-1. \qquad \text{.....(A9)}$$

APPENDIX B <u>Reed Solomon Code</u>

An example of Reed Solomon code is shown in Fig. B1, where one word (symbol) consists of three bits. Two check word P and Q are generated as solutions of the following equation.

$$HV = 0 \qquad \text{.....(B1)}$$

where;

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ \alpha^6 & \alpha^5 & \alpha^4 & \alpha^3 & \alpha^2 & \alpha \end{bmatrix} \qquad \text{.....(B2)}$$

$$V = \begin{bmatrix} A \\ B \\ C \\ D \\ P \\ Q \end{bmatrix} \qquad \text{.....(B3)}$$

$\alpha^6 \sim \alpha$ are considered as weighting factors just like $T^4 \sim T$ in b-Adjacent code, but the execution of multiplication is completely different in Reed solomon code, because the product is defined on Galois field.

The α is called as primitive element and is defined as a solution of the following formula (modulo two), for instance.

$$F(x) = x^3+x+1 = 0 \qquad \text{.....(B4)}$$

$$\therefore \quad \alpha^3+\alpha+1 = 0 \qquad \text{.....(B5)}$$

Usually each bit of three bit word is expressed as x^2, x, or 1 as follows.

$$x^2 = (100) \qquad \text{.....(B4)}$$

$$x = (010) \qquad \text{.....(B7)}$$

$$1 - (001) \qquad \text{.....(B8)}$$

$$x^2+1 = (101) \qquad \text{.....(B9)}$$

$\vdots$

Assuming,

$$\alpha = x \qquad \text{.....(B10)}$$

then

$$\alpha^2 = x^2 = (100) \qquad \text{.....(B11)}$$

$$\alpha^3 = \alpha+1 = x+1 = (011) \qquad \text{.....(B12)}$$

$$\alpha^4 = \alpha\cdot\alpha^3 = \alpha(\alpha+1) = \alpha^2+\alpha = (110) \qquad \text{.....(B13)}$$

$$\alpha^5 = \alpha^2+\alpha+1 = (111) \qquad \text{.....(B14)}$$

$$\begin{aligned}\alpha^6 &= \alpha\cdot\alpha^5 = \alpha(\alpha^2+\alpha+1) = \alpha^3+\alpha^2+\alpha \\ &= \alpha+1+\alpha^2+\alpha = \alpha^2+1 = (101)\end{aligned} \qquad \text{.....(B15)}$$

$$\begin{aligned}\alpha^7 &= \alpha(\alpha^2+1) = \alpha^3+\alpha \\ &= \alpha+1+\alpha = 1 = (001)\end{aligned} \qquad \text{.....(B16)}$$

(Note; in modulo two, $1+1 = \alpha+\alpha = \alpha^2+\alpha^2 = 0$)

Hence all possible word of three bits, (000)~(111), are expressed by elements (0, $\alpha^7 = 1, \alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6$). From this, product is defined as follows.

$$\alpha\cdot\alpha = \alpha^2,\ (010)\ (010) = (100) \qquad \text{.....(B17)}$$

$$\alpha\cdot\alpha^2 = \alpha^3,\ (010)\ (100) = (011) \qquad \text{.....(B18)}$$

$$\alpha^2\cdot\alpha^3 = \alpha^5,\ (100)\ (011) = (111) \qquad \text{.....(B19)}$$

$$\alpha^2\cdot\alpha^5 = \alpha^7 = 1,\ (100)\ (111) = (001) \qquad \text{.....(B20)}$$

$$1\cdot\alpha^2 = \alpha^2,\ (001)\ (100) = (100) \qquad \text{.....(B21)}$$

$\vdots$

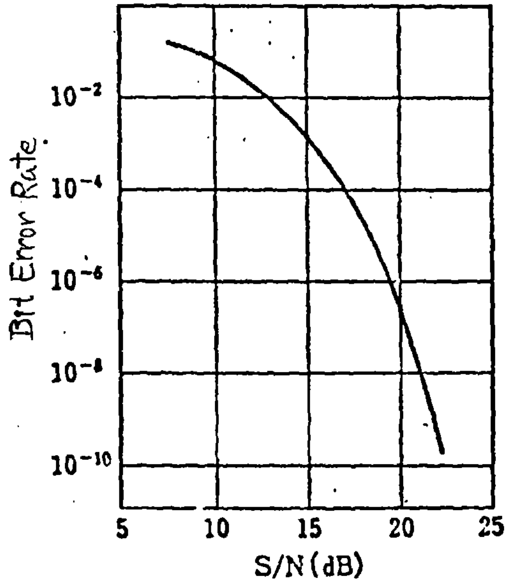
The all possible product laws are shown in Fig. B2. The Galois field is defined as a group of finite elements, which satisfies commutative law, arrociative law, distributive law and inverse of element (i.e. α^{-1}, α^{-2}, ...).

Using these laws, the equation (B2) is solved as follows.

$$P = \alpha A + \alpha^2 B + \alpha^5 C + \alpha^3 D \qquad \text{.....(B22)}$$

$$Q = \alpha^3 A + \alpha^6 B + \alpha^4 C + \alpha D \qquad \text{.....(B23)}$$

Single word error correction is possible because the ratio of weight facters on each word is different. Double erasure can be correted because two equations exist for two unknown values. the excecution of correction is similar to that of b-Adjacent code but the circuit is simpler in Reed Solomon code by unique laws of multiplication shown in Fig. B2.



S = Signal (p-p)

N = Noise (rms)

Fig. 2.1 Error Rate and S/N

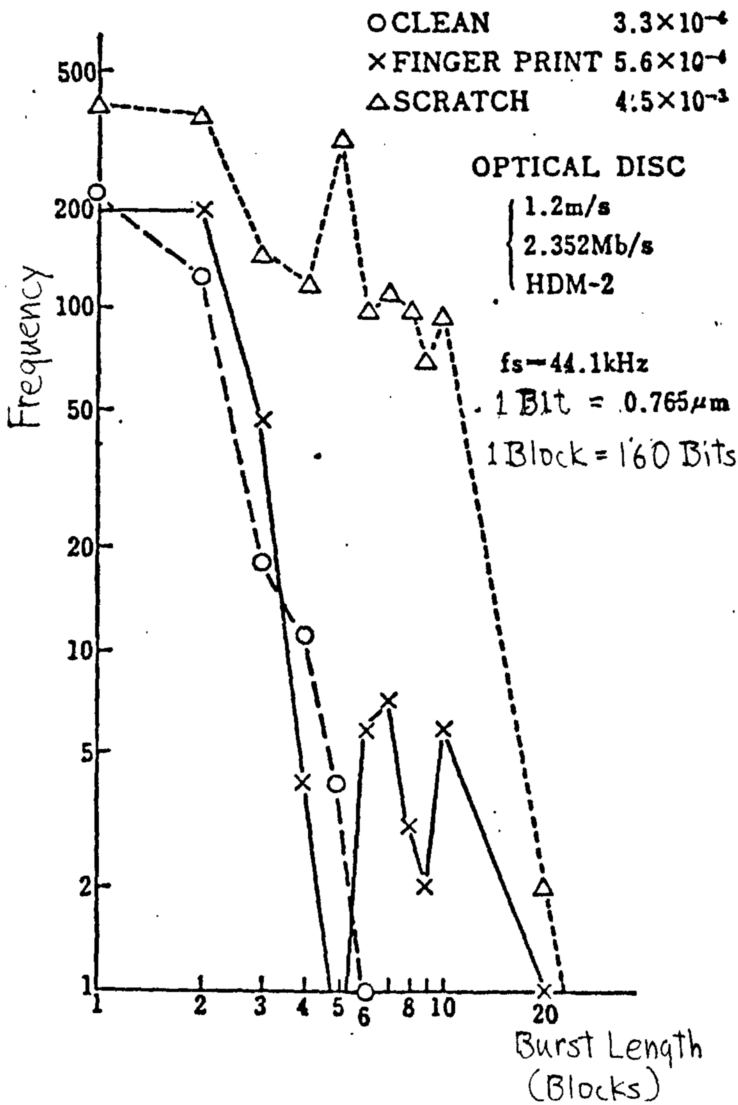
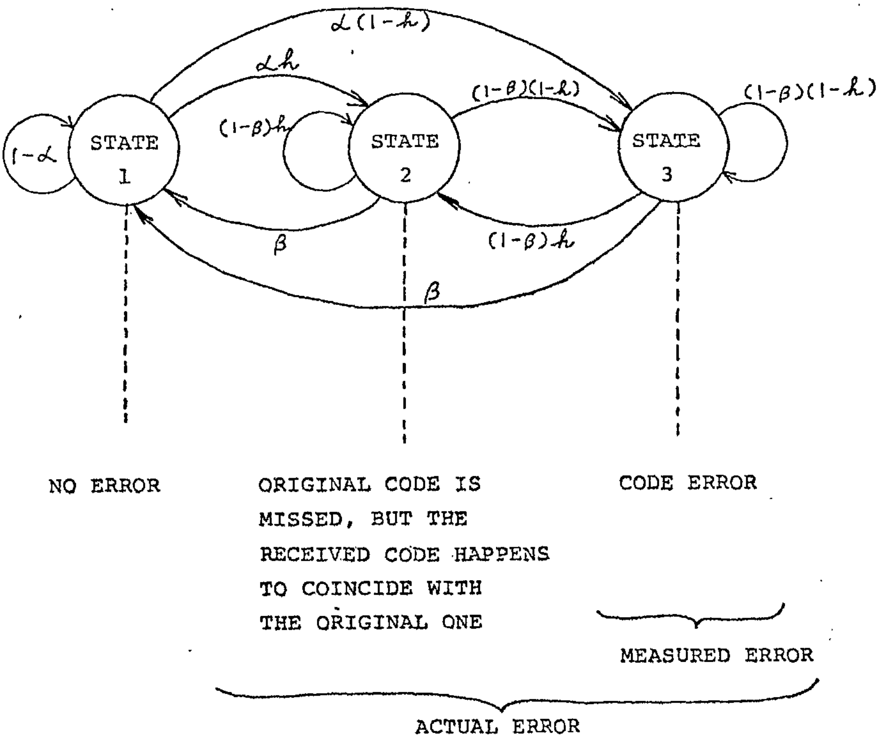


Fig. 2.2 An Example of Error Statistics



α :PROBABILITY TO FALL INTO ERROR STATE 2,3
β :PROBABILITY TO RECOVER FROM ERROR STATE 2,3
h :PROBABILITY OF ERRONEOUS CODE ACCIDENTALLY COINCIDE WITH ORIGINAL ONE

Fig-2.3 A STATISTICAL MODEL OF CODE ERRORS (MODIFIED GILBERT)

Fig. 3.1 Mis-Detection Probability of CRCC
Block Length = 128 bit
Bit Error Rate = 10^{-3}

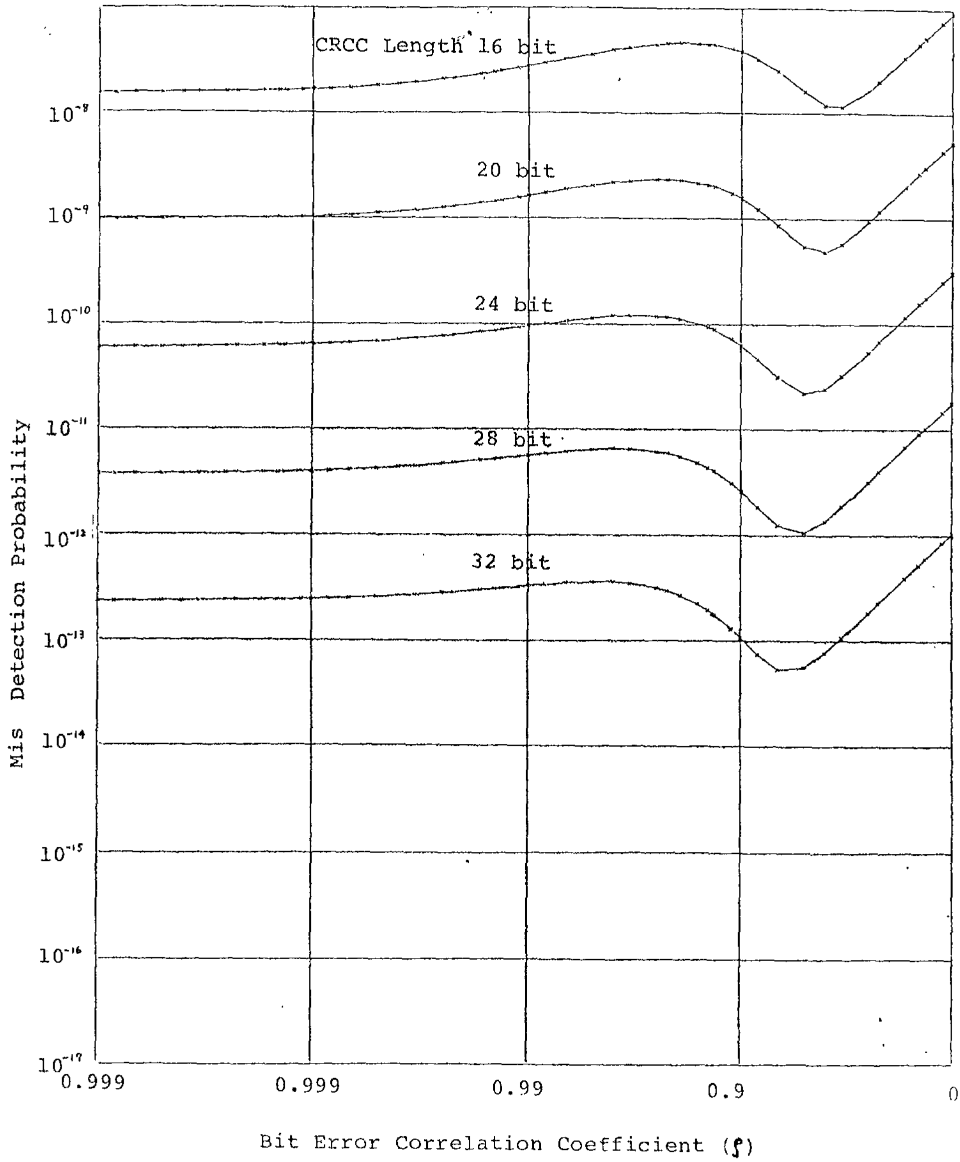


Fig. 3.2 Mis-Detection Probability of CRCC

Block Length = 256 bit
Bit Error Rate = 10^{-3}

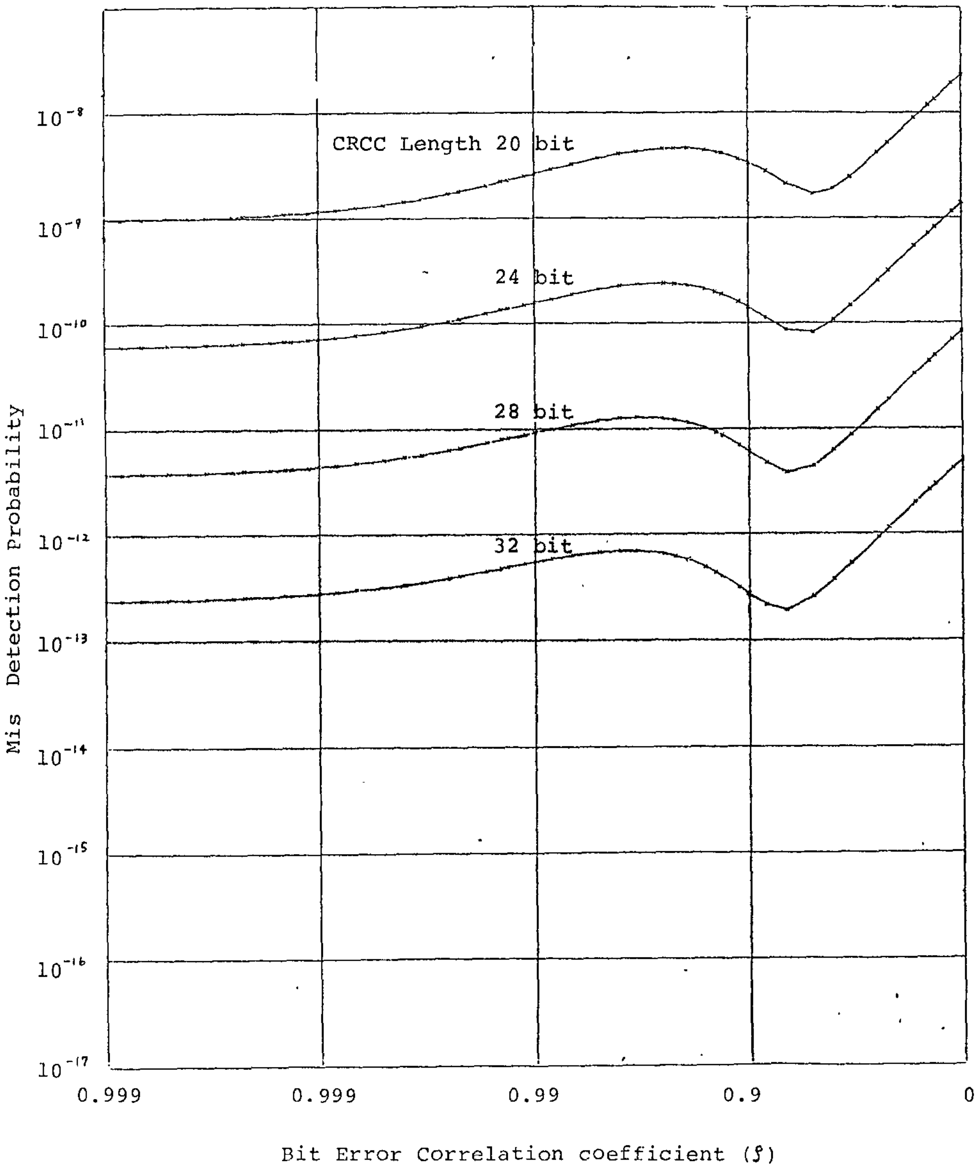
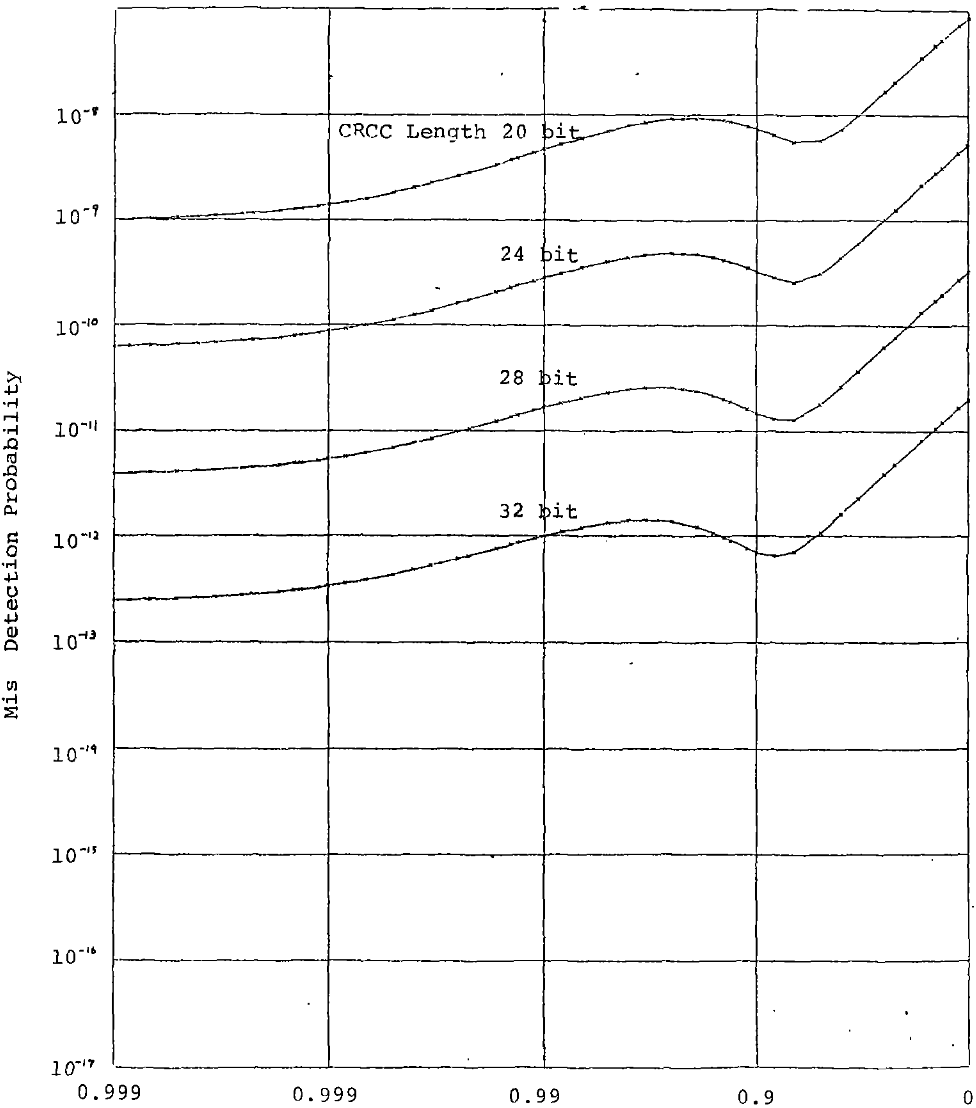


Fig. 3.3 Mis - Detection Probability of CRCC

Block Length = 512 bit
Bit Error Rate = 10^{-3}



Bit Error Correlation Coefficient (ρ)

Fig. 3.4 Mis - Detection Probability of CRCC

Block Length = 128 bit
Bit Error Rate = 10^{-4}

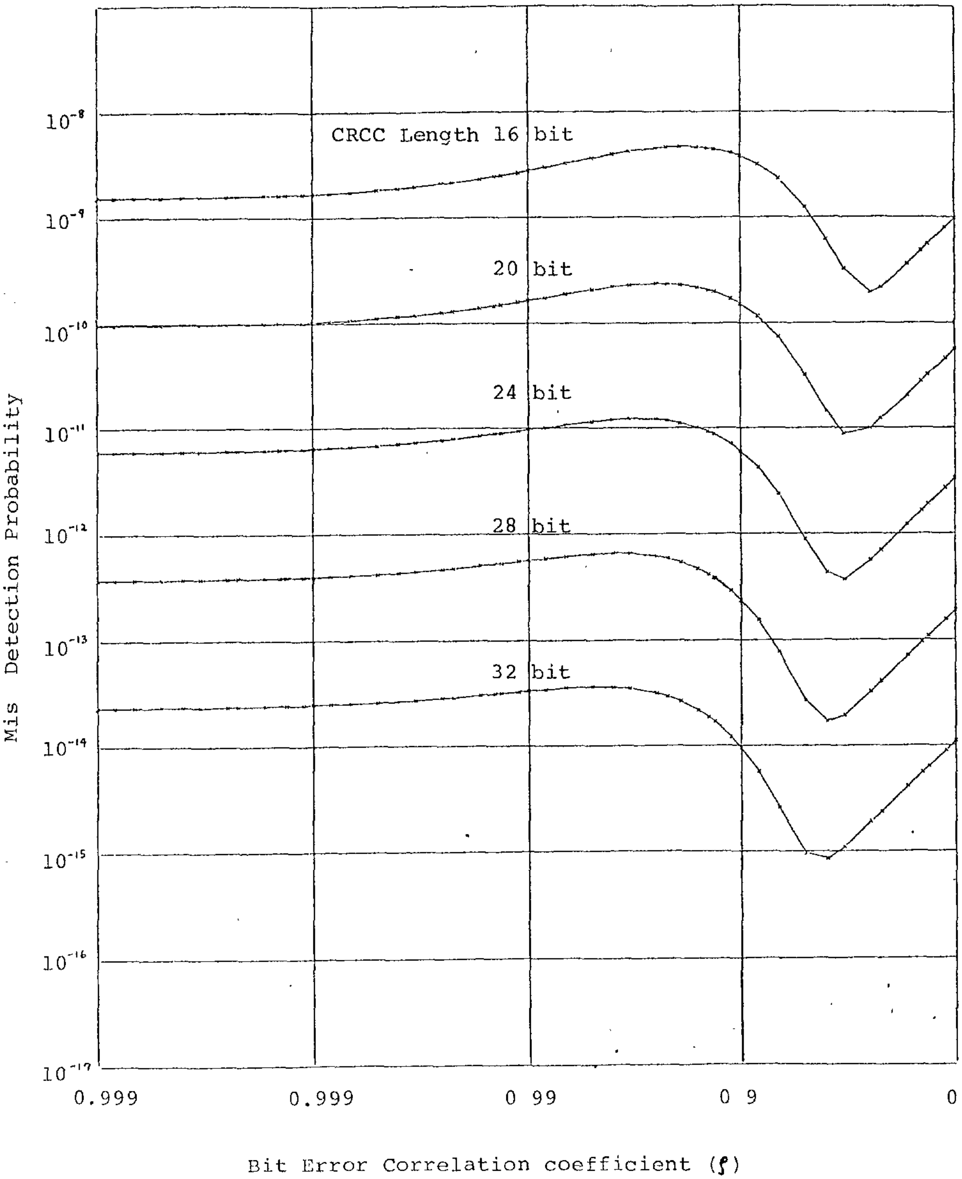


Fig. 3.5 Mis - Detection Probability of CRCC

Block Length = 256 bit
Bit Error Rate = 10^{-4}

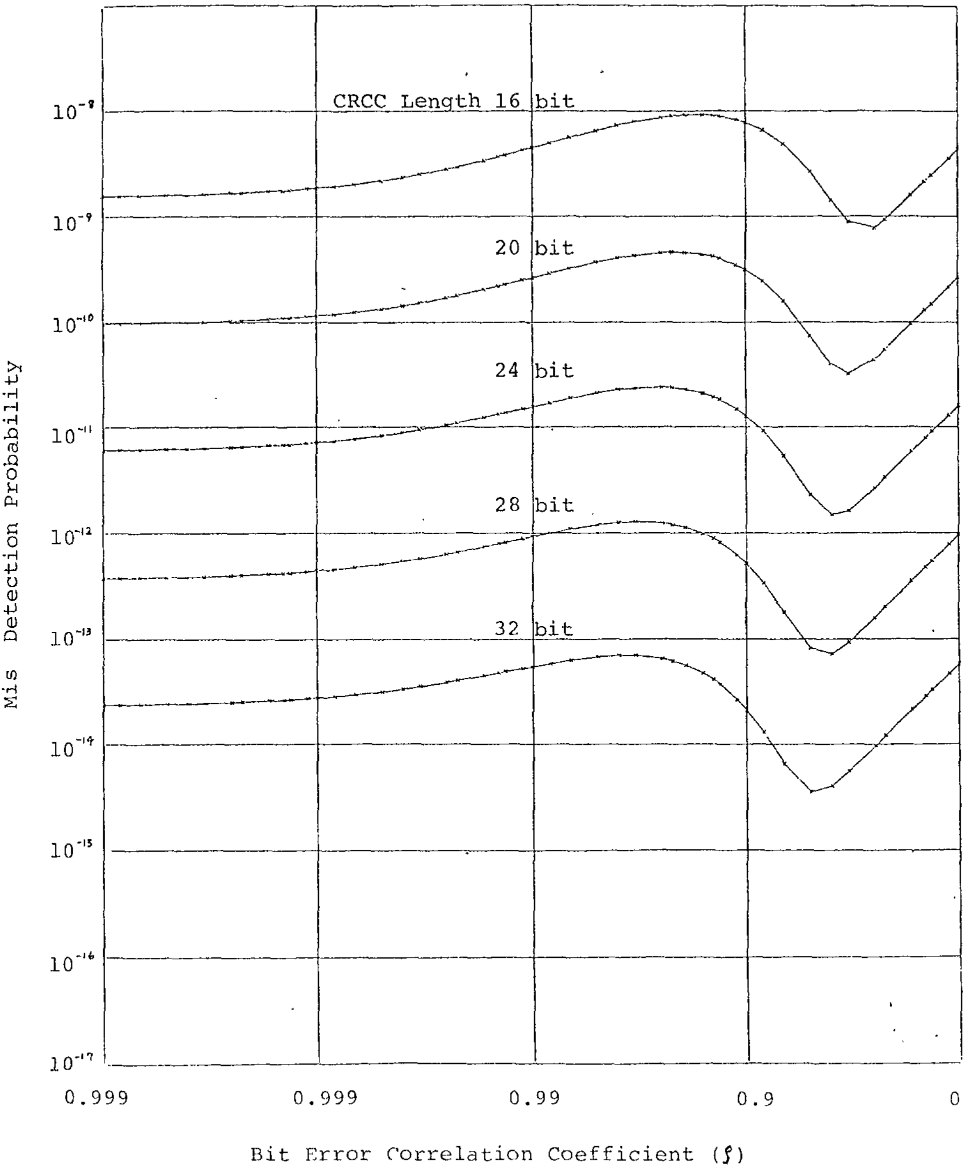
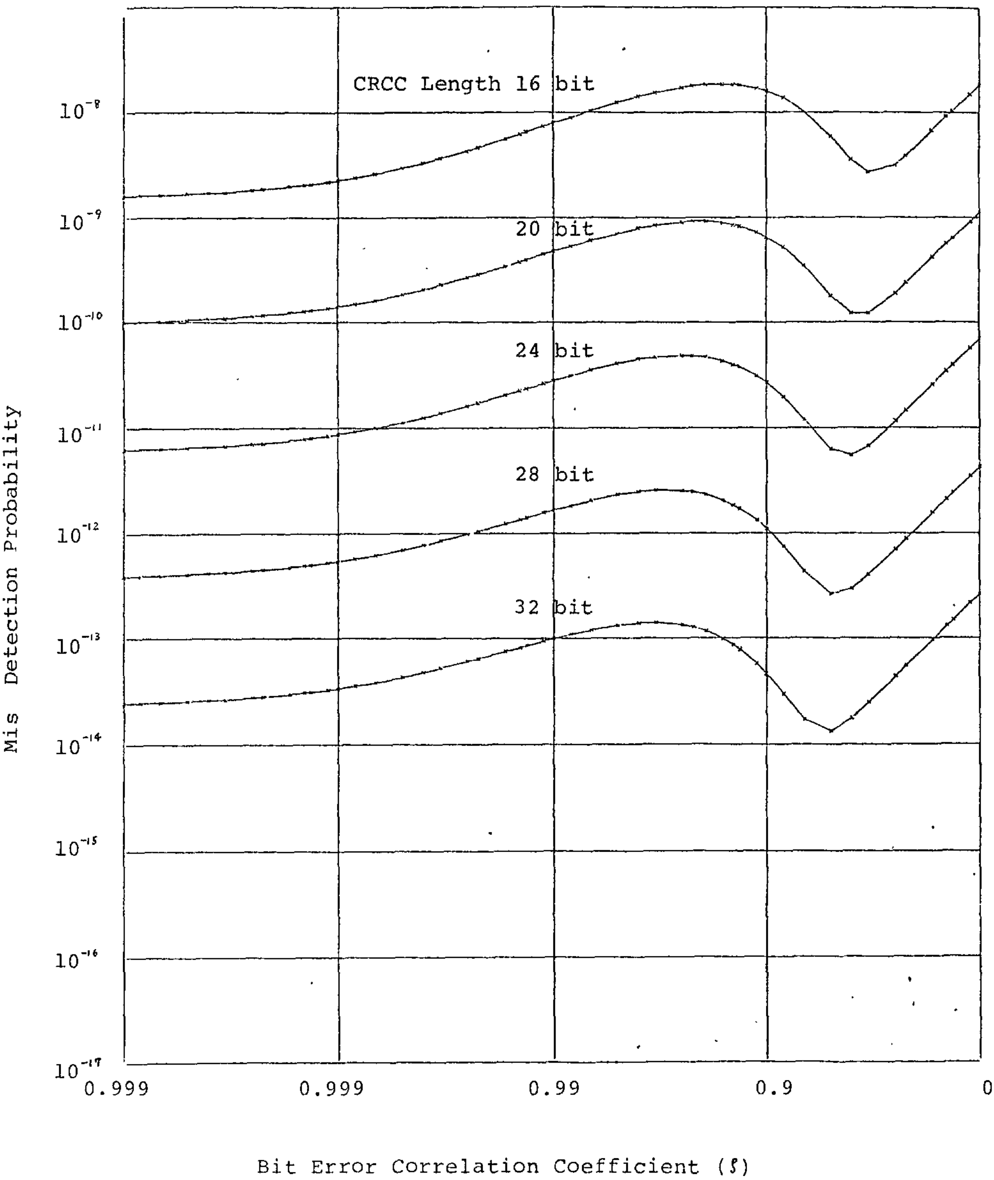


Fig. 3.6 Mis- Detection Probability of CRCC

Block Length = 512 bit
Bit Error Rate = 10^{-4}



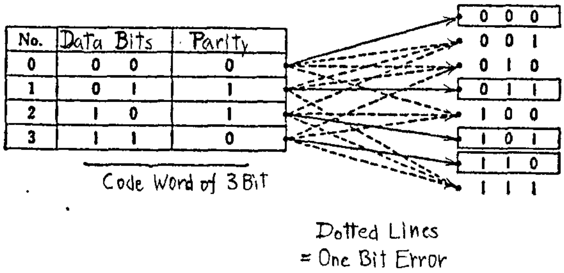


Fig.4.1 One Bit Error Detection Code

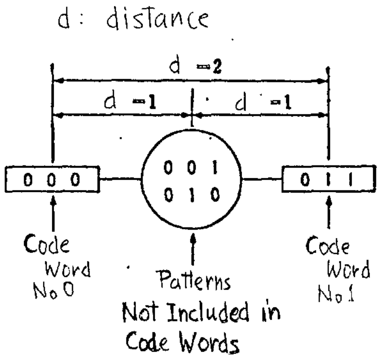


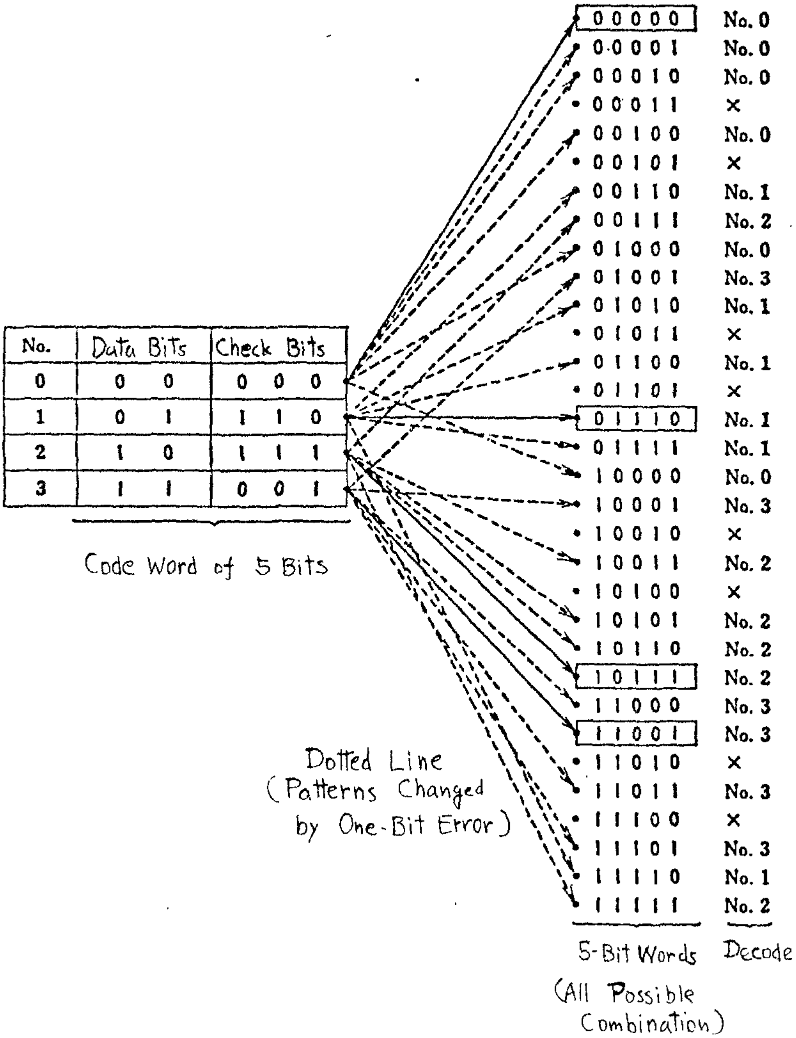
Fig. 4.2 Distance d = 2

No of Word	Data Bit	Parity Bit
0	X 0	0
1	X 1	1
2	X 0	1
3	X 1	0

Erasure

Fig. 4,3 Single Erasure

Fig. 4.4 One Bit Error Correction



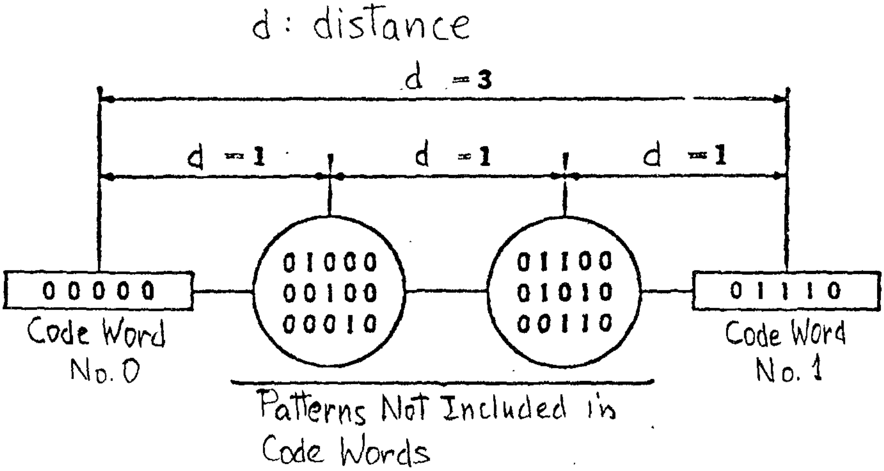


Fig 4.5 Distance d = 3

SLIP (A)
(Original slip)

W_1	\$ 100.-
W_2	\$ 200.-
W_3	\$ 300.-
W_4	\$ 400.-
P(Total)	\$1,000.-

Syndrome (without error)

$S=W_1+W_2+W_3+W_4-P=0$

SLIP (B)
(one word erasure)

W_1	\$ 100.-
W_2^*	\$
W_3	\$ 300.-
W_4	\$ 400.-
P(Total)	\$1,000.-

Syndrome (with one erasure)

$S=W_1+W_2^*+W_3+W_4-P=-200$

Erasure Correction
$W_2=W_2^*-S=200$

SLIP (C)
(one word error)

W_1'	\$ 100.-
W_2'	\$ 300.-
W_3'	\$ 300.-
W_4'	\$ 400.-
P'(Total)	\$1,000.-

Syndrome

$S=W_1+W_2+W_3+W_4-P=100$

SLIP (D)
(one word error with pointer)

W_1	\$ 100.-	
W_2^*	\$ 300.-	← Error Pointer
W_3	\$ 300.-	
W_4	\$ 400.-	
P'(Total)	\$1,000.-	

Syndrome S=100

Correction
$W_2=W_2^*-S=200$

Fig. 5,1 The Principle of Single Erasure Correction

SLIP(E) (original slip)	
W_1	\$ 100.-
W_2	\$ 200.-
W_3	\$ 300.-
W_4	\$ 400.-
P(Total)	\$1,000.-
Q(Weighted Total)	\$3,000.-

SLIP(F) (single error)	
W_1'	\$ 100.-
W_2'	\$ 300.-
W_3'	\$ 300.-
W_4'	\$ 400.-
P'(Total)	\$1,000.-
Q'(Weighted Total)	\$2,000.-

$P=W_1+W_2+W_3+W_4$

$Q=4W_1+3W_2+2W_3+W_4$

Syndrome(without error)

$S_1=W_1+W_2+W_3+W_4-P=0$

$S_2=4W_1+3W_2+2W_3+W_4-Q=0$

Syndrome

$S_1=W_1'+W_2'+W_3'+W_4'-P'=100$

$S_2=4W_1'+3W_2'+2W_3'+W_4'-Q'=300$

for Single Error		
	then	
if $4S_1=S_2$	, W_1'	is erroneous
" $3S_1=S_2$	, W_2'	"
" $2S_1=S_2$	, W_3'	"
" $S_1=S_2$	, W_4'	"
" $S_1\neq 0, S_2=0$	, P'	"
" $S_1=0, S_2\neq 0$	, Q'	"

W_2' is corrected.

Fig. 6,1 Single Error Correction Code

SLIP (G)

W_1	\$ 100.-	Error
W_2^*	\$ 300.-	Pointer
W_3^*	\$ 400.-	
W_4	\$ 400.-	
P(Total)	\$1,000.-	
Q(Weighted Total)	\$2,000.-	

Syndrome

$S_1 = W_1 + W_2^* + W_3^* + W_4 - P = 200$

$S_2 + 4W_1 + 3W_2^* + 2W_3^* + W - Q = 500$

$W_2^* = W_2 + E_2$

$W_3^* = W_3 + E_3$

$S_1 = E_2 + E_3 = 200$

$S_2 = 3E_2 + 2E_3 = 500$

$\therefore \quad E_2 = S_2 - 2S_1 = 100$

$\quad\;\; E_3 = 3S_1 - S_2 = 100$

Correction

$W_2 = W_2^* - E_2 = 200$

$W_3 = W_3^* - E_3 = 300$

Fig. 6,2 Double Erasure Correction

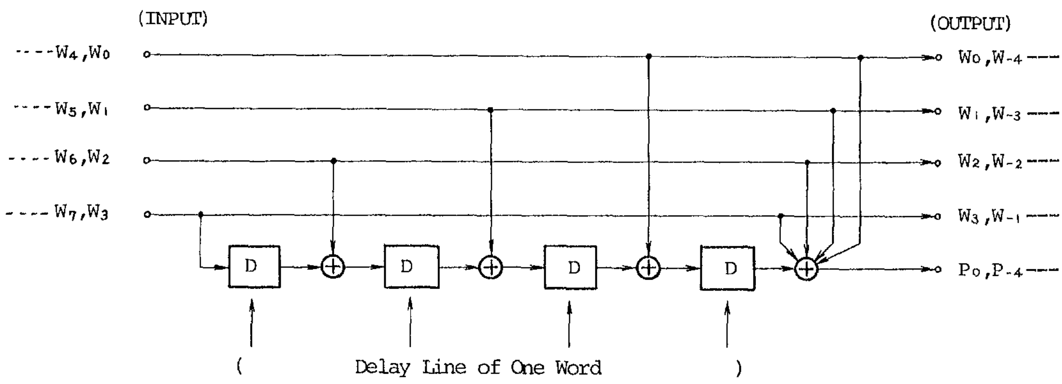


Fig. 7,1 An Example of Convolutional Code (Encoder)

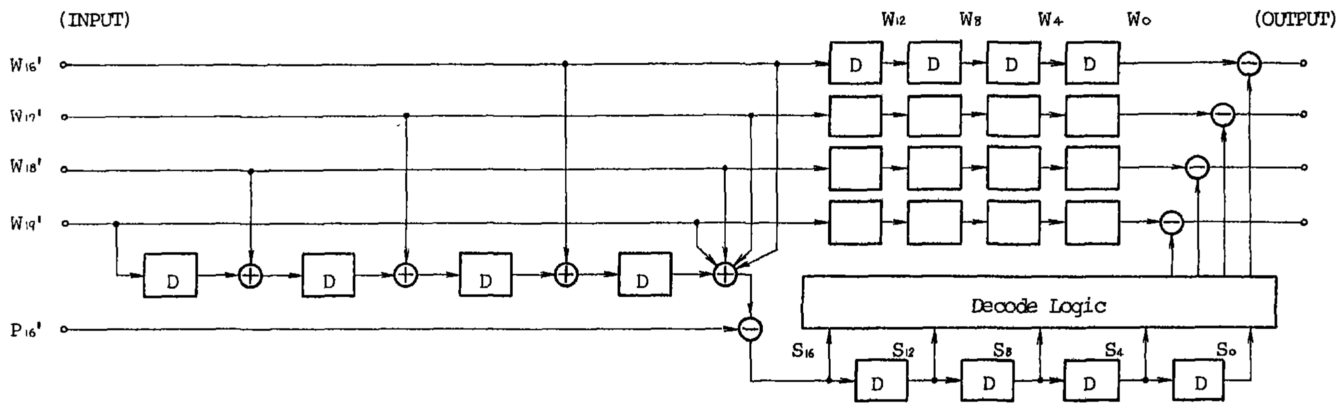


Fig. 7,2 A Convolutional Code (Decoder)

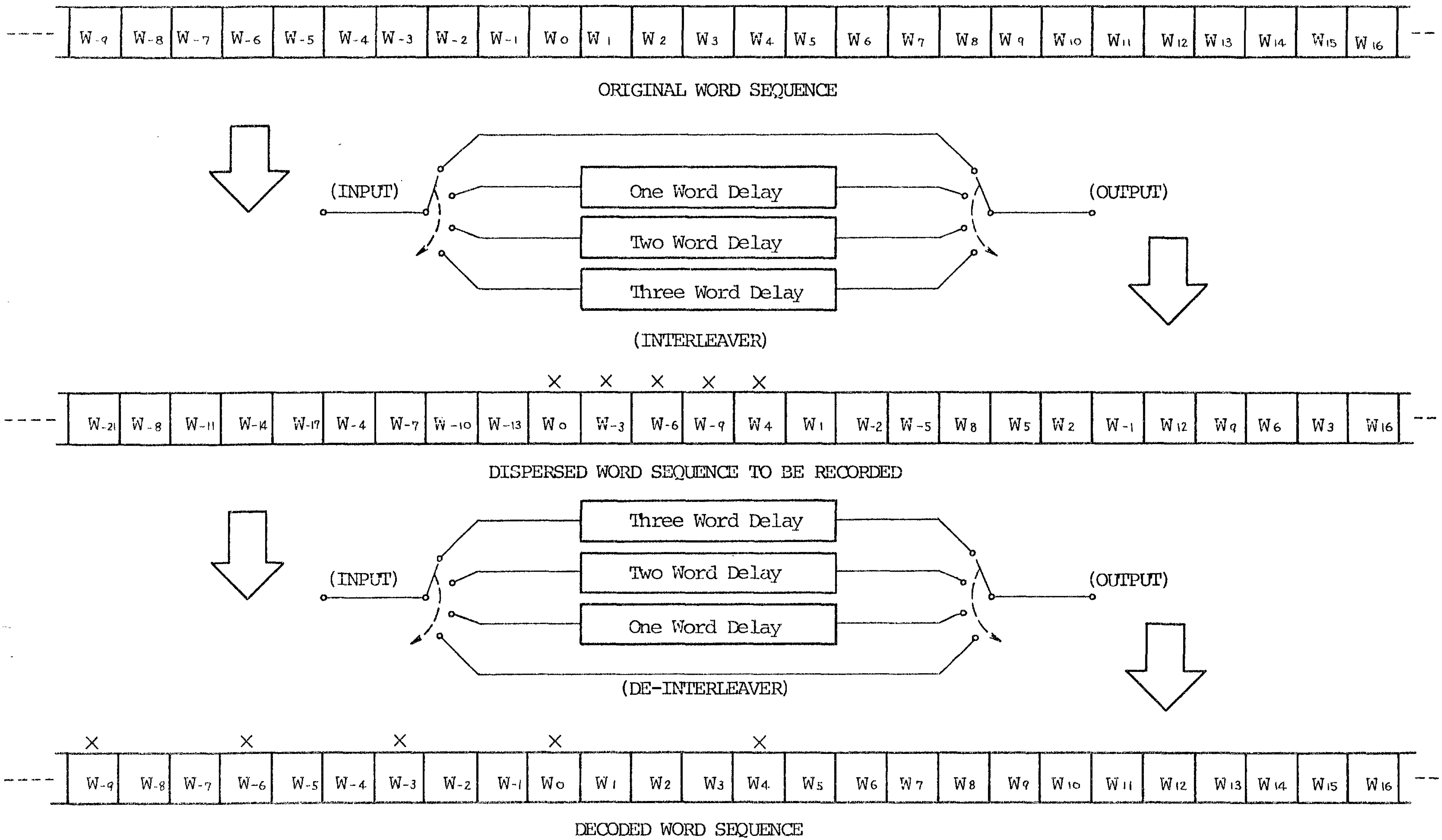
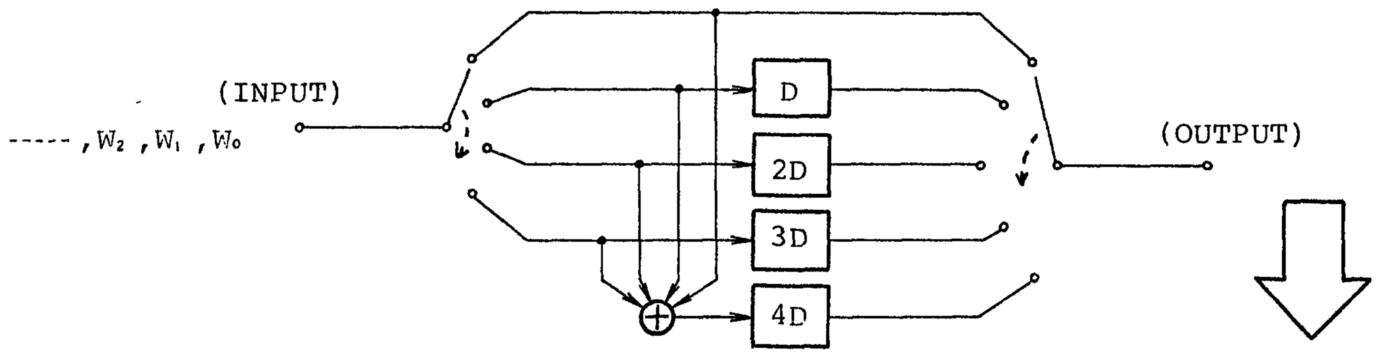
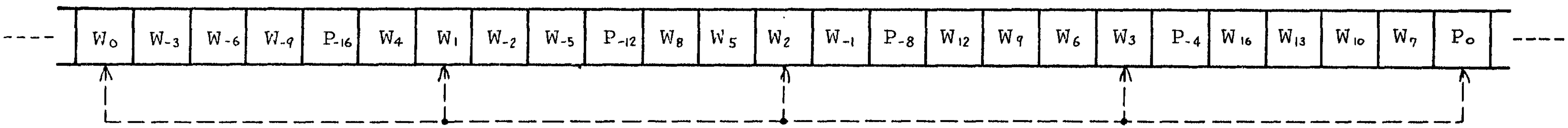


Fig. 8,1 Interleave and De-Interleave



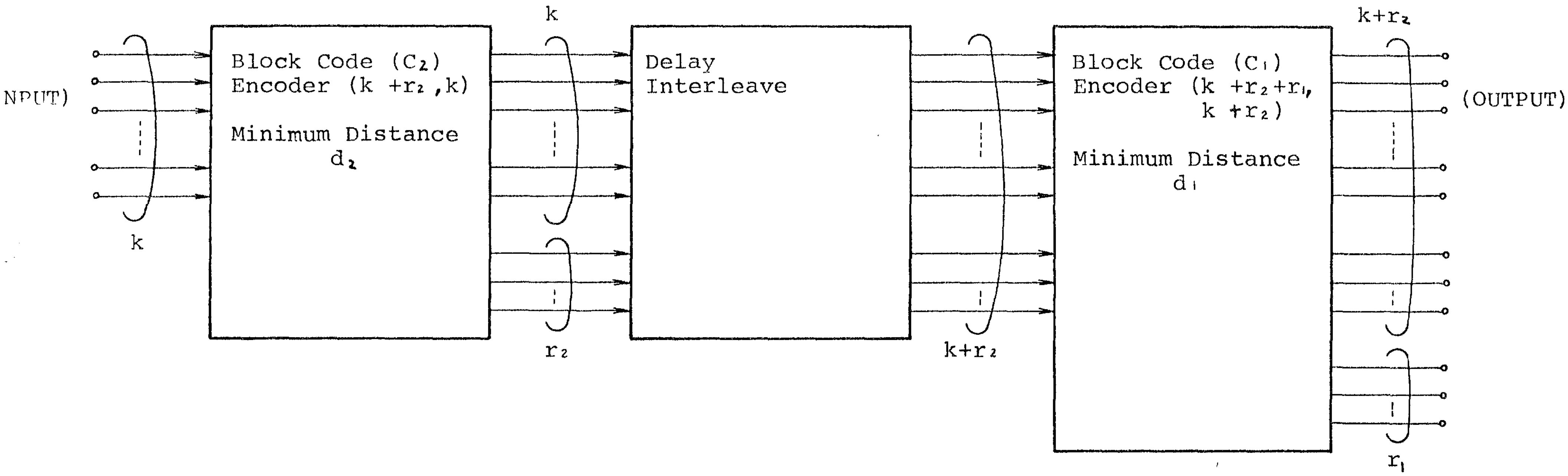
$P_0 = W_0 + W_1 + W_2 + W_3$
$P_4 = W_4 + W_5 + W_6 + W_7$
⋮

DISPERSED WORD SEQUENCE TO BE RECORDED (D = 1 Word)



Related Words of Original Block Code

Fig. 8.2 Single Erasure correction Code with Interleave



Original Data Words: k

Check Words for C_2 : r_2

Check Words for C_1 : r_1

Total Check Words : r_1+r_2

Redundancy : $(r_1+r_2)/(k+r_1+r_2)$

Fig. 8.3 Cross Interleave Method (Encoder)

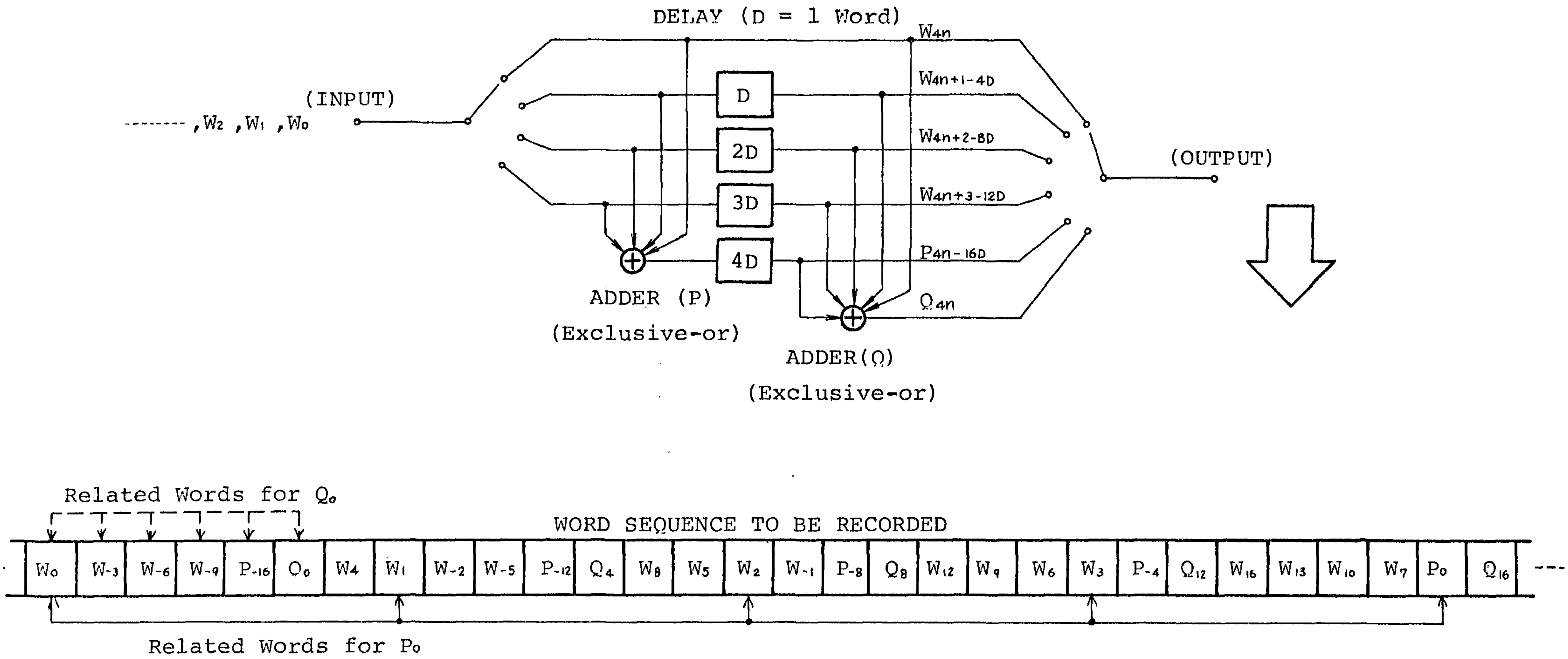


Fig. 8.4 An Example of CIC Encoder

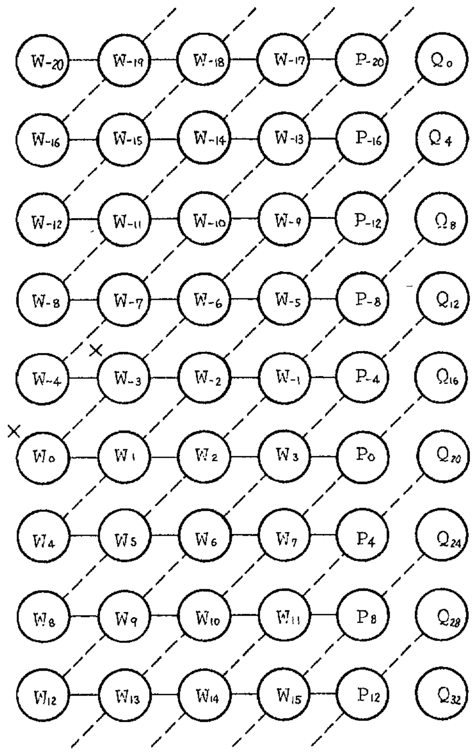


Fig. 8.5 Code Diagram of CIC

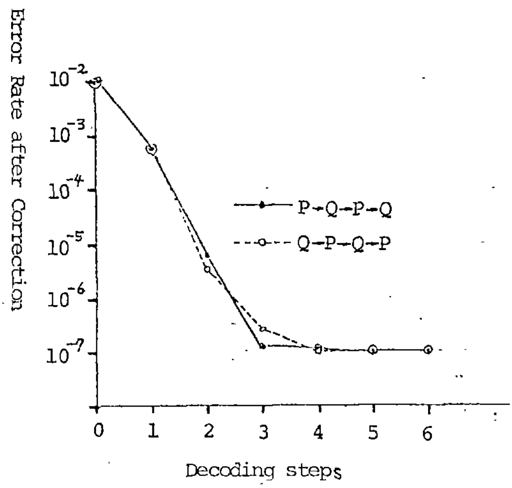


Fig. 8,6 Correctability of CIC against Random Word Error

One Block = 6 Data Words + 2 Check Words

Word Error Rate : Pw = 10^{-2}

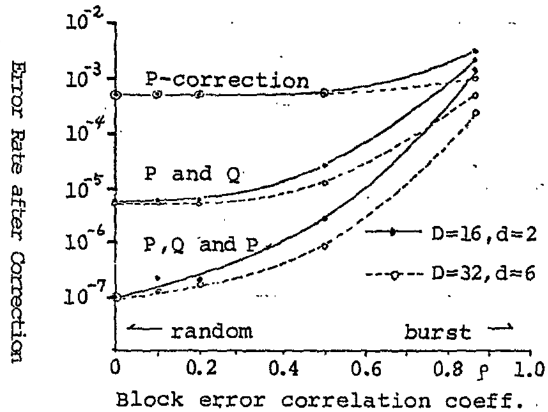


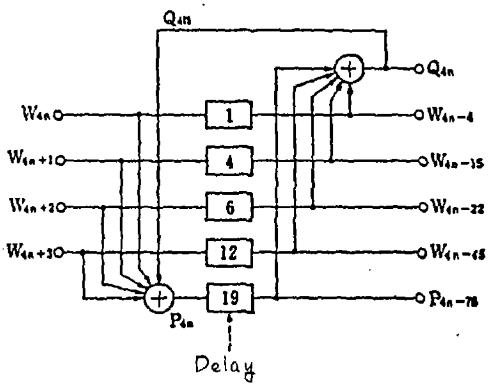
Fig. 8,7 Correctability of CIC against Burst Error

One Block = 6 Data Words + 2 Check Words

Word Error Rate : $P_W = 10^{-2}$

Unit Delay Inside CIC Encoder : d

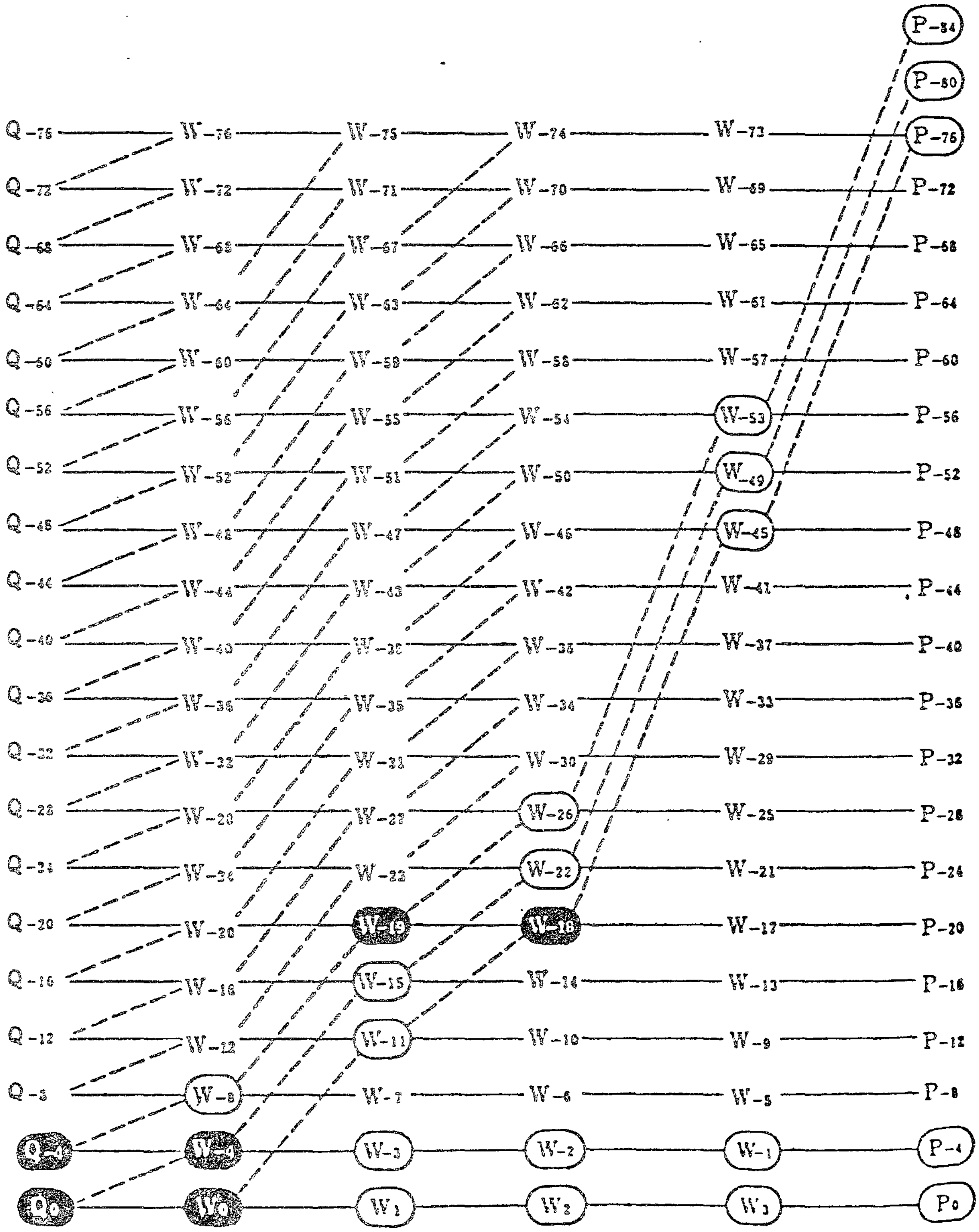
Unit Delay for Interleaving after CIC Encoder : D



$$P_{4n} = Q_{4n} + W_{4n} + W_{4n+1} + W_{4n+2} + W_{4n+3}$$
$$Q_{4n} = P_{4n-76} + W_{4n-4} + W_{4n-15} + W_{4n-22} + W_{4n-45}$$

Fig 8.8 An Example of ICIC Encoder

Fig. 8.9 Code Diagram for ICIC



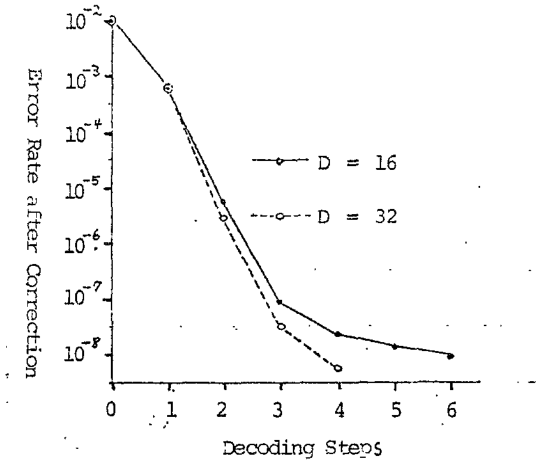


Fig. 8,10 Correctability of ICIC against Random Word Error

One Block = 6 Data Words + 2 Check Words

Word Error Rate : $P_w = 10^{-2}$

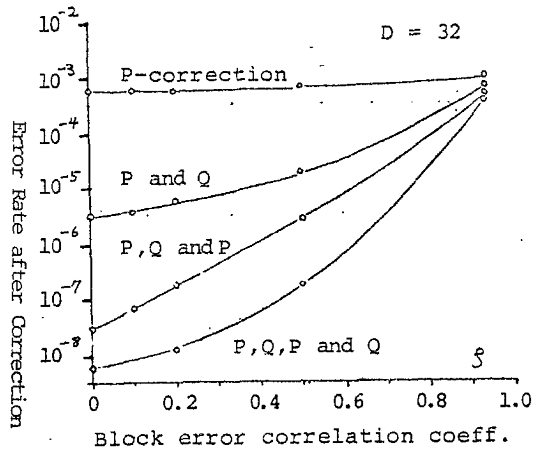


Fig. 8,11 Correctability of ICIC against Burst Error

One Block = 6 Data Words + 2 Check Words

Word Error Rate : $Pw = 10^{-2}$

Unit Delay for Interleaving after CIC Encoder : D = 32

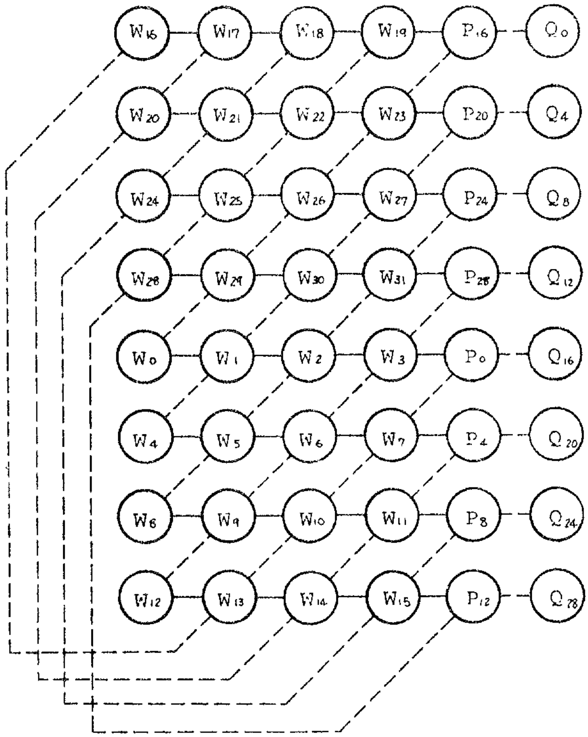


Fig. 8.12 Block Completed CIC

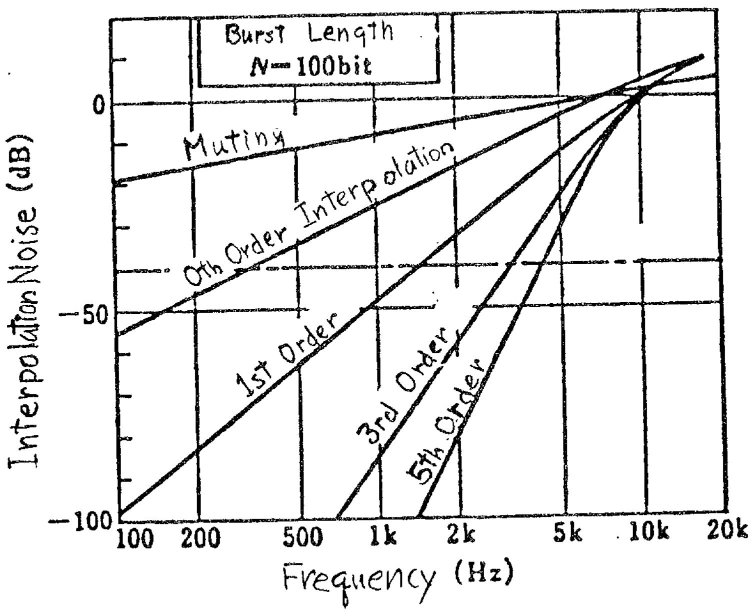


Fig 9.1 Interpolation Noise

OUTLINE OF SUBJECTIVE TEST

(1) Number of subject = 8 (men)
(2) Signal = pure tone
(3) The minimum detectable frequency is shown, when the subject makes interpolation at arbitrary timings.
(4) Experiment 1,2,and 3 correspond to the number of good words (0) between interpolated words (x); one, two, and three respectively.
(5) $\bar{x}$ = mean value
σ = standard deviation

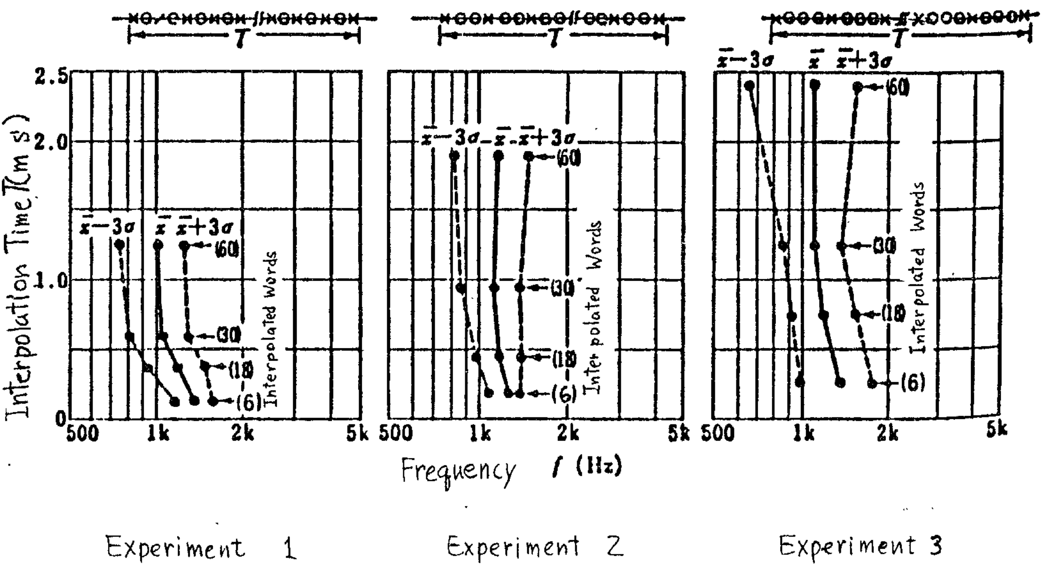


Fig. 9.2 Subjective Test of Interpolation

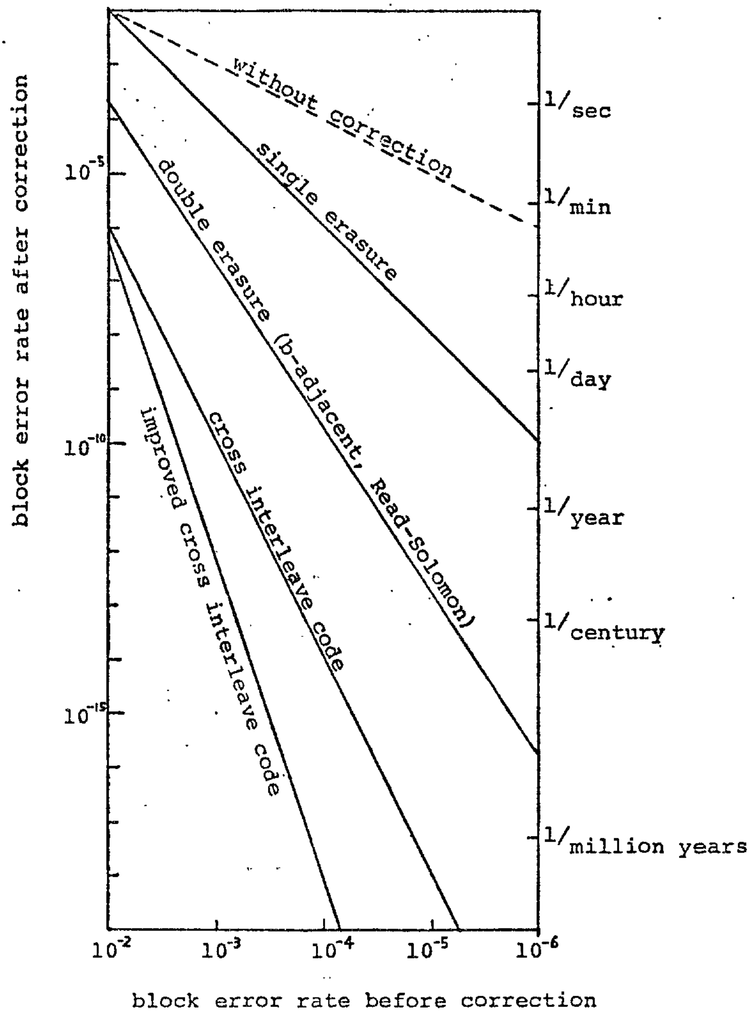


Fig. 10.1 Random error correctability of various codes

region (i) : good to use, block error rate after correction $<10^{-8}$
region (ii) : warning , $10^{-8}<$ 〃 $<10^{-4}$
region (iii) : inhibited to use, $10^{-4}<$ 〃

area A : best tuned condition
area B : deteriorated in studio environment

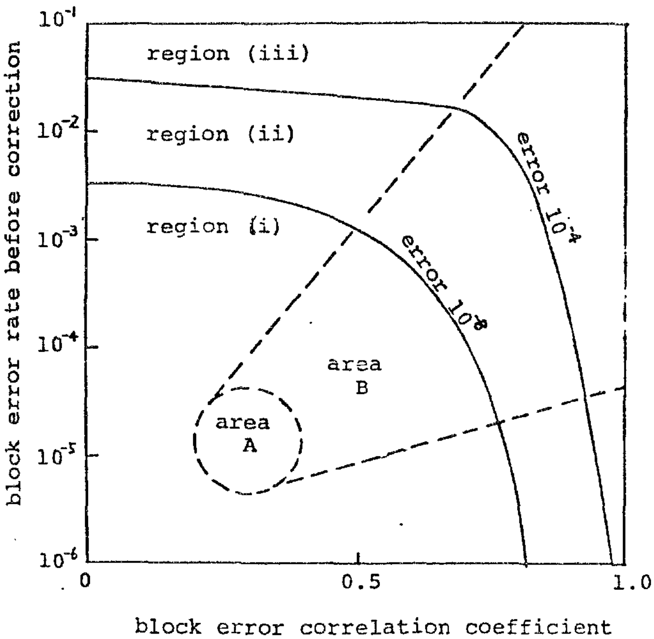


Fig. 10.2 Evaluation of "DASH Format"

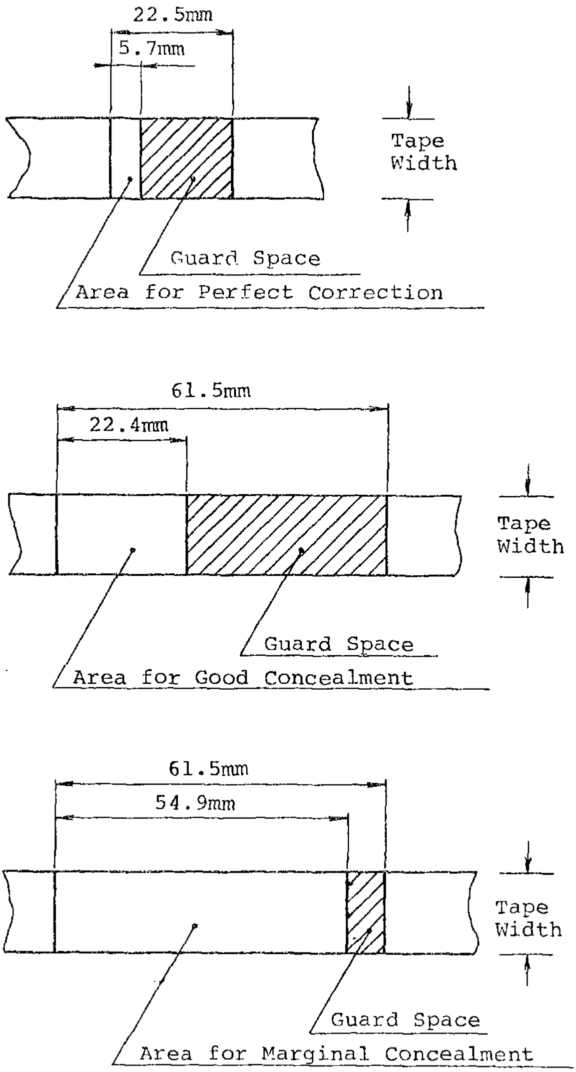


Fig. 10.3 Guard Space of DASH Format

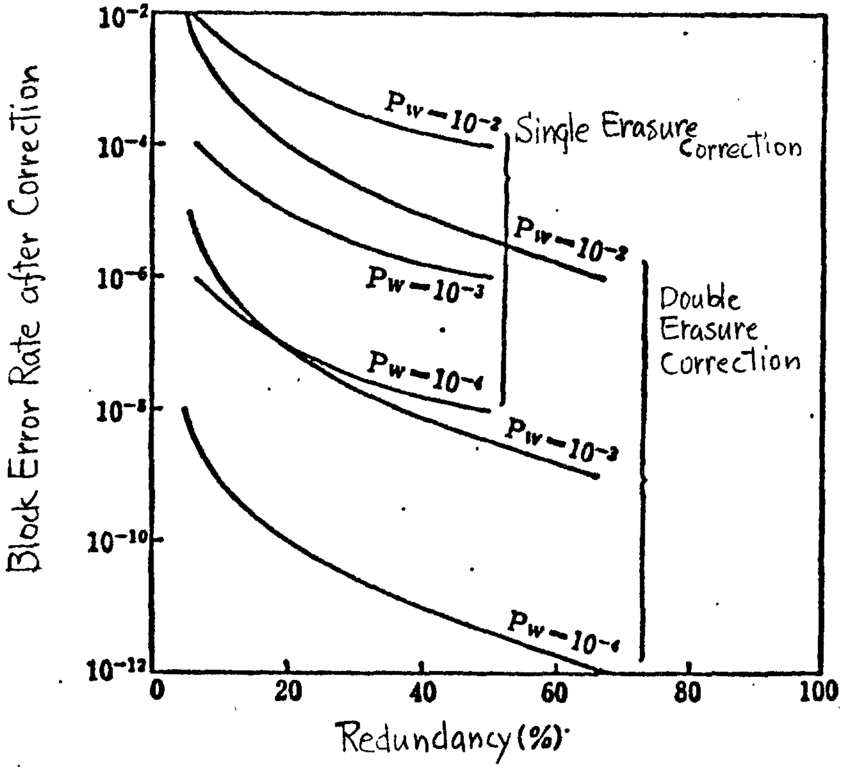


Fig. 10.4 Redundancy and Correctability

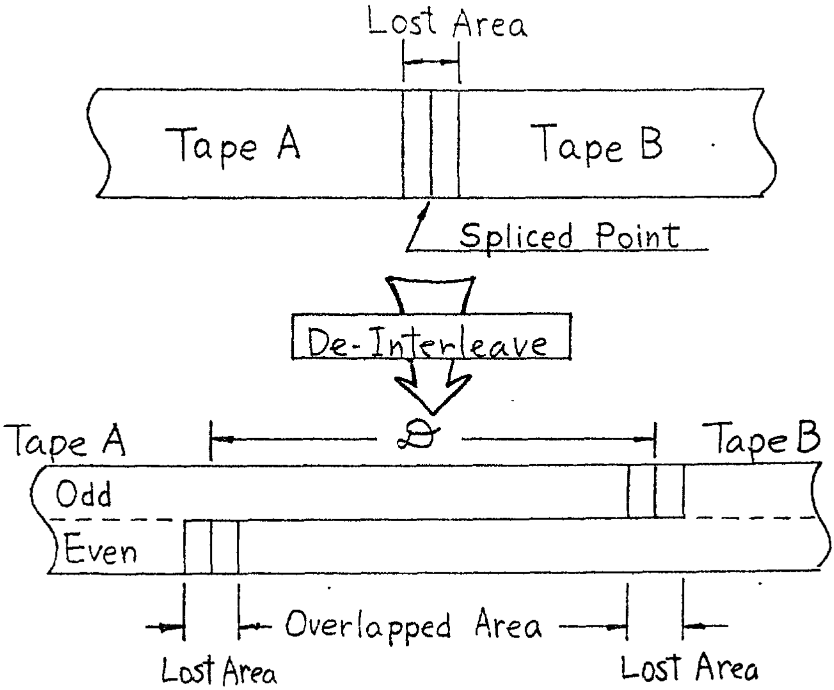
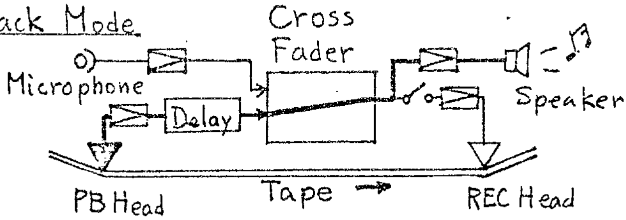


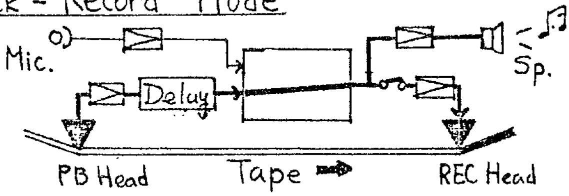
Fig. 10.5 Tape Splice Editing

Fig.10.6 The Procedure for Punching In/Out

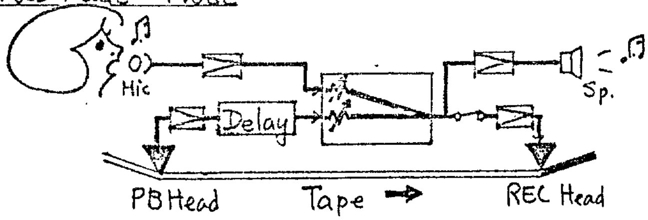
(i) Playback Mode



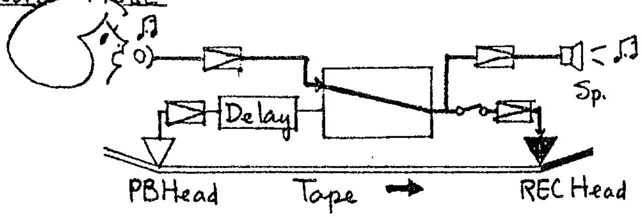
(ii) Playback - Record Mode



(iii) Cross Fade Mode



(iv) Record Mode



(v) Cross Fade Mode

(vi) Playback - Record Mode

(vii) Playback Mode

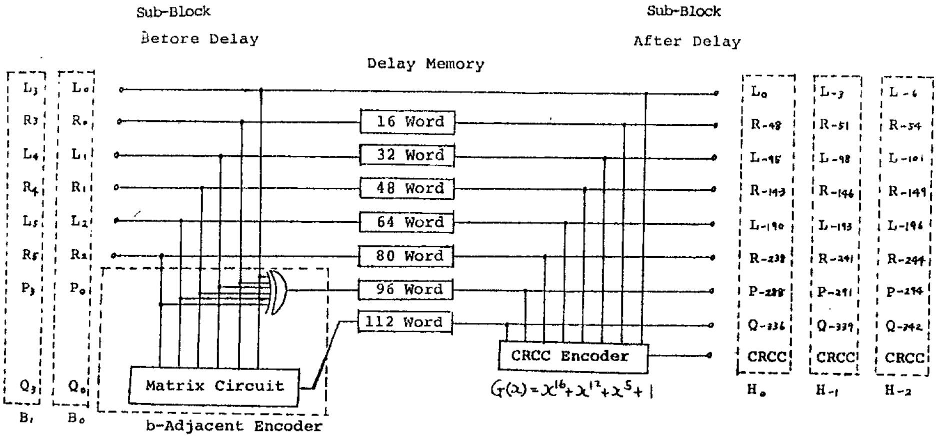


Fig-11.1 Encoder

$$P_n = L_n \oplus R_n \oplus L_{n+1} \oplus R_{n+1} \oplus L_{n+2} \oplus R_{n+2}$$

$$Q_n = T^6 \cdot L_n \oplus T^5 \cdot R_n \oplus T^4 \cdot L_{n+1} \oplus T^3 \cdot R_{n+1} \oplus T^2 \cdot L_{n+2} \oplus T \cdot R_{n+2}$$

where $\oplus$ means modulo 2 summation. (Exclusive OR)

$$T = \begin{pmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&1\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&1\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0
\end{pmatrix}$$

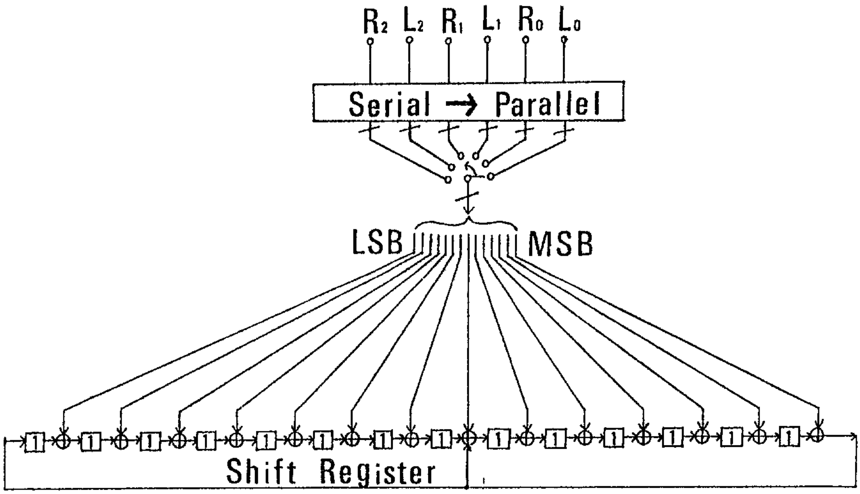


Fig. II.2 Matrix Circuit for b-Adjacent Encoding

H_0	L_0	R_{-48}	L_{-95}	R_{-143}	L_{-190}	R_{-238}	P_{-288}	Q_{-336}	CRCC
H_{16}	L_{48}	R_0	L_{-47}	R_{-95}	L_{-142}	R_{-190}	P_{-240}	Q_{-288}	CRCC
H_{32}	L_{96}	R_{48}	L_1	R_{-47}	L_{-94}	R_{-42}	P_{-192}	Q_{-240}	CRCC
H_{48}	L_{144}	R_{96}	L_{49}	R_1	L_{-46}	R_{-94}	P_{-144}	Q_{-192}	CRCC
H_{64}	L_{192}	R_{144}	L_{97}	R_{49}	L_2	R_{-46}	P_{-96}	Q_{-144}	CRCC
H_{80}	L_{240}	R_{192}	L_{145}	R_{97}	L_{50}	R_2	P_{-48}	Q_{-96}	CRCC
H_{96}	L_{288}	R_{240}	L_{193}	R_{145}	L_{98}	R_{50}	P_0	Q_{-48}	CRCC
H_{112}	L_{336}	R_{288}	L_{241}	R_{193}	L_{146}	R_{98}	P_{48}	Q_0	CRCC

Fig. 11.3 Data Sequence on Tape

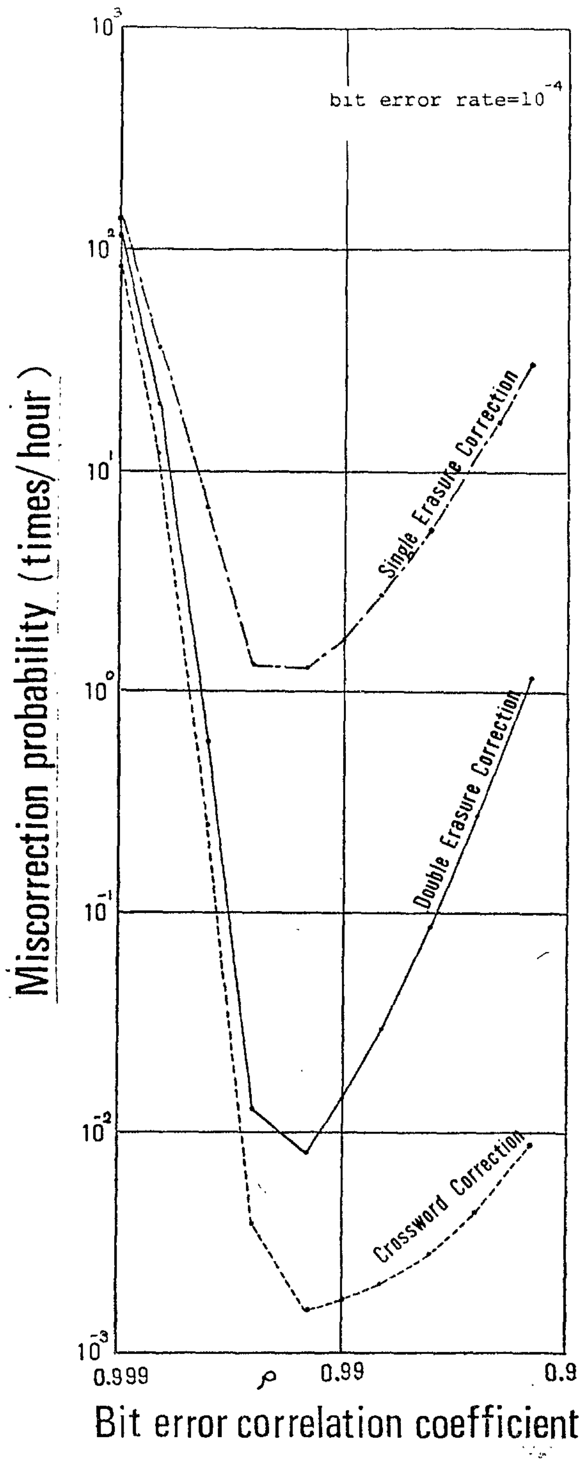


Fig-11.4 Correctability of EIAJ Format

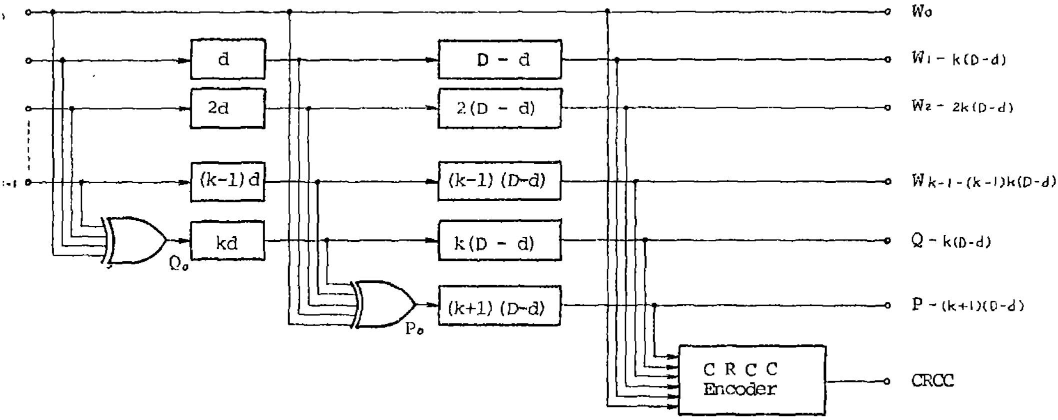


Fig. 12,1 CIC (Cross Interleave Code) with Delay Interleave (Encoder)

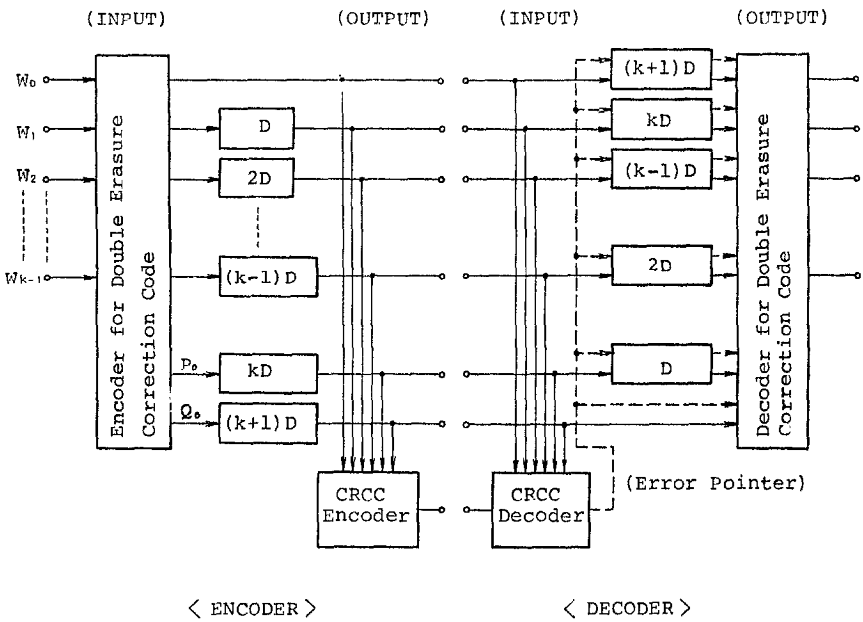


Fig. 12.2 Double Erasure Correction Code with Delay Interleave

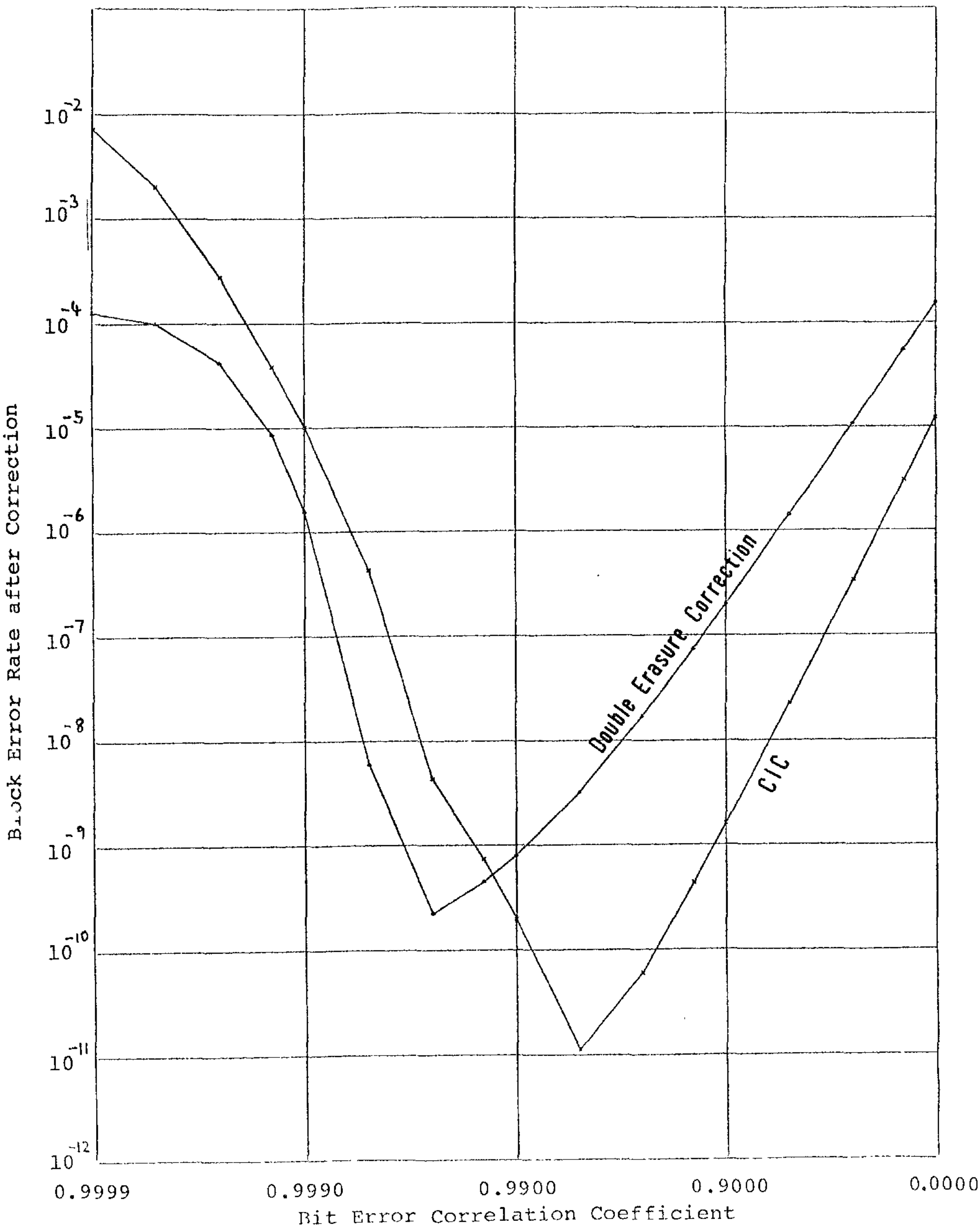


Fig. 12.3 CIC and Double Erasure Correction (1)
k=6, D=10, d=1, Bit Error Rate=10^{-4}

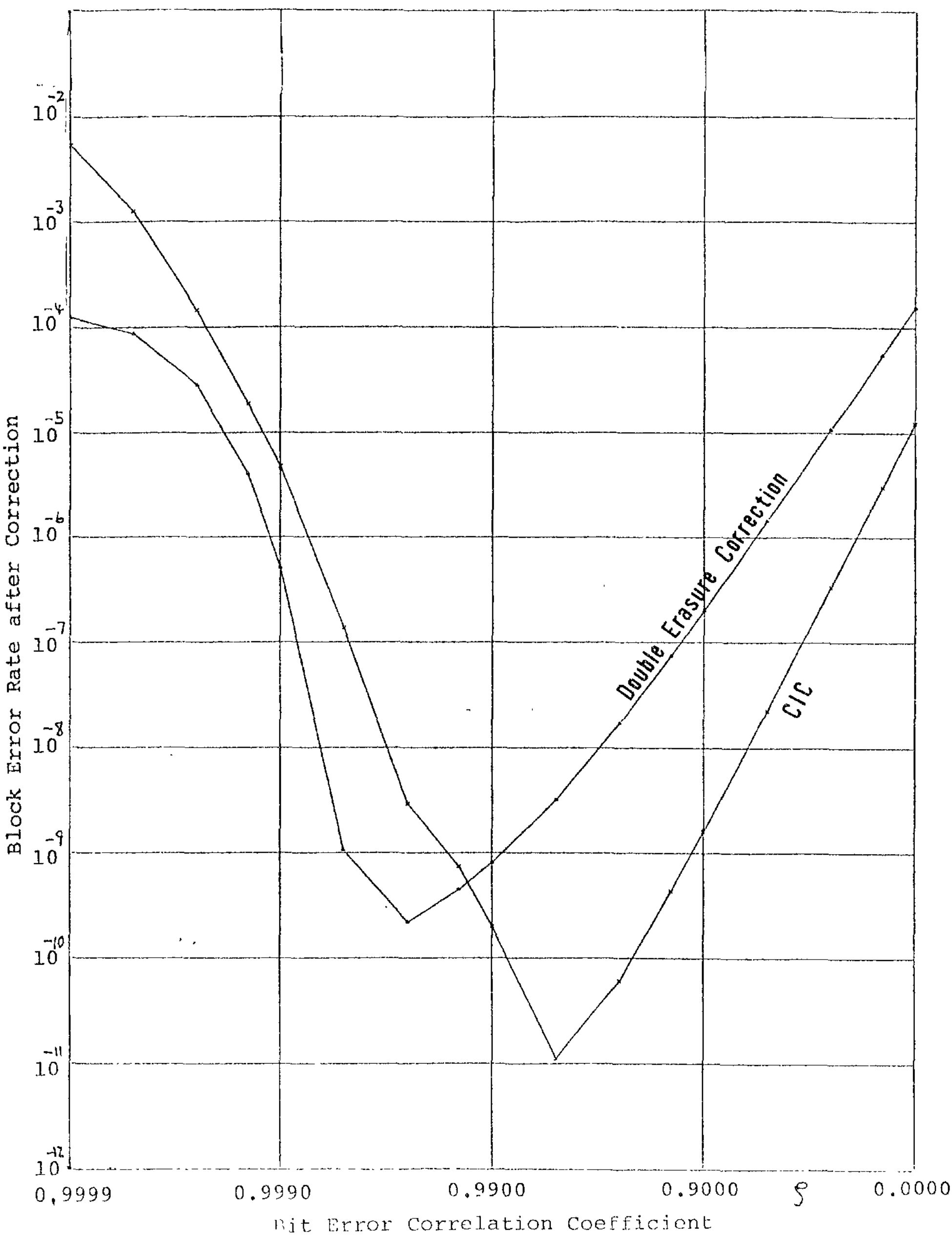


Fig. 12.4 CIC and Double Erasure Correciton (2)

k=6, D=12, d=1, Bit Error Rate=10^{-4}

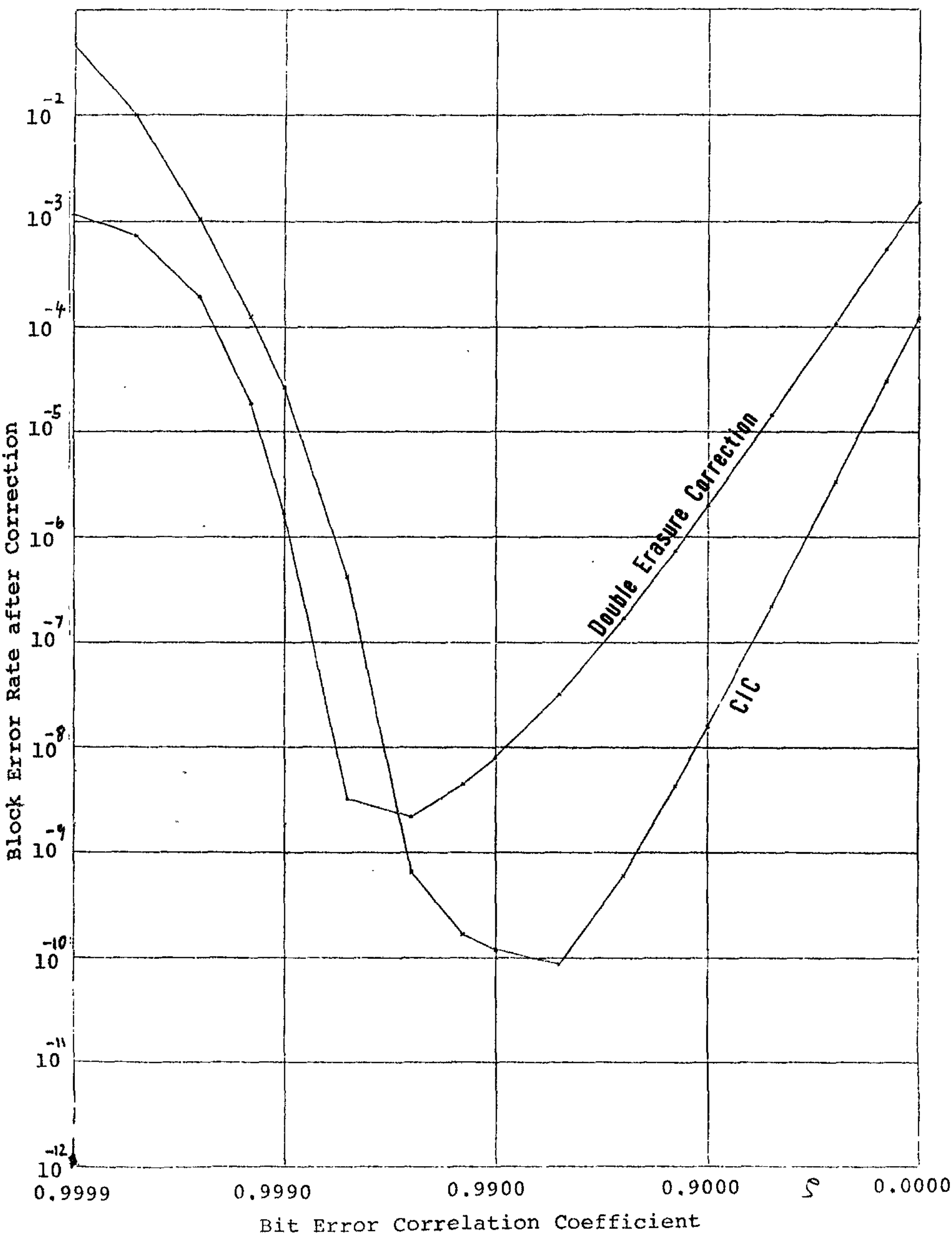


Fig. 12.5 CIC and Double Erasure Correction (3)
k=6, D=14, d=3, Bit Error Rate=10^{-4}

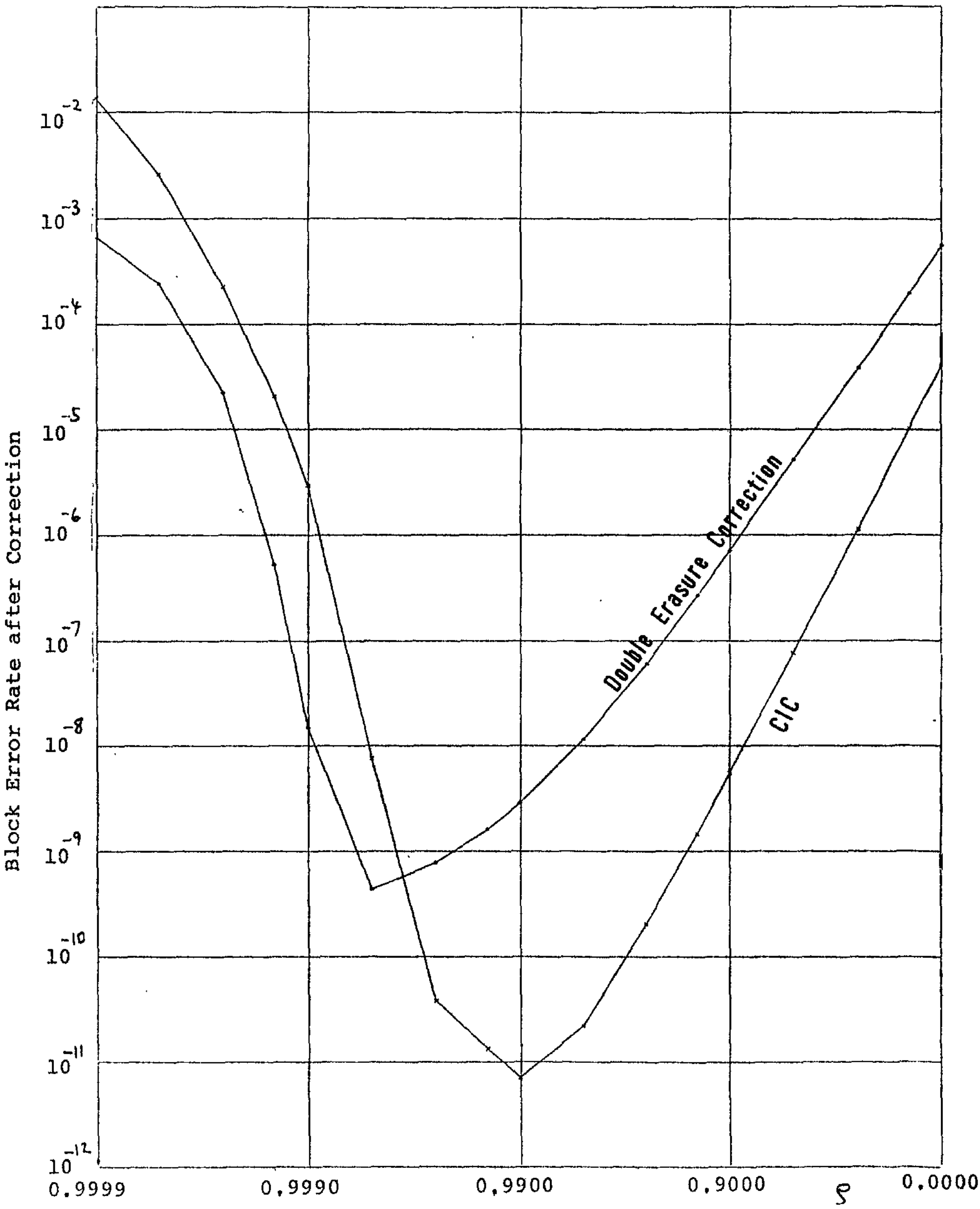


Fig. 12.6 CIC and Double Erasure Correction (4)
$k=4$, $D=22$, $d=6$, Bit Error Rate$=10^{-4}$

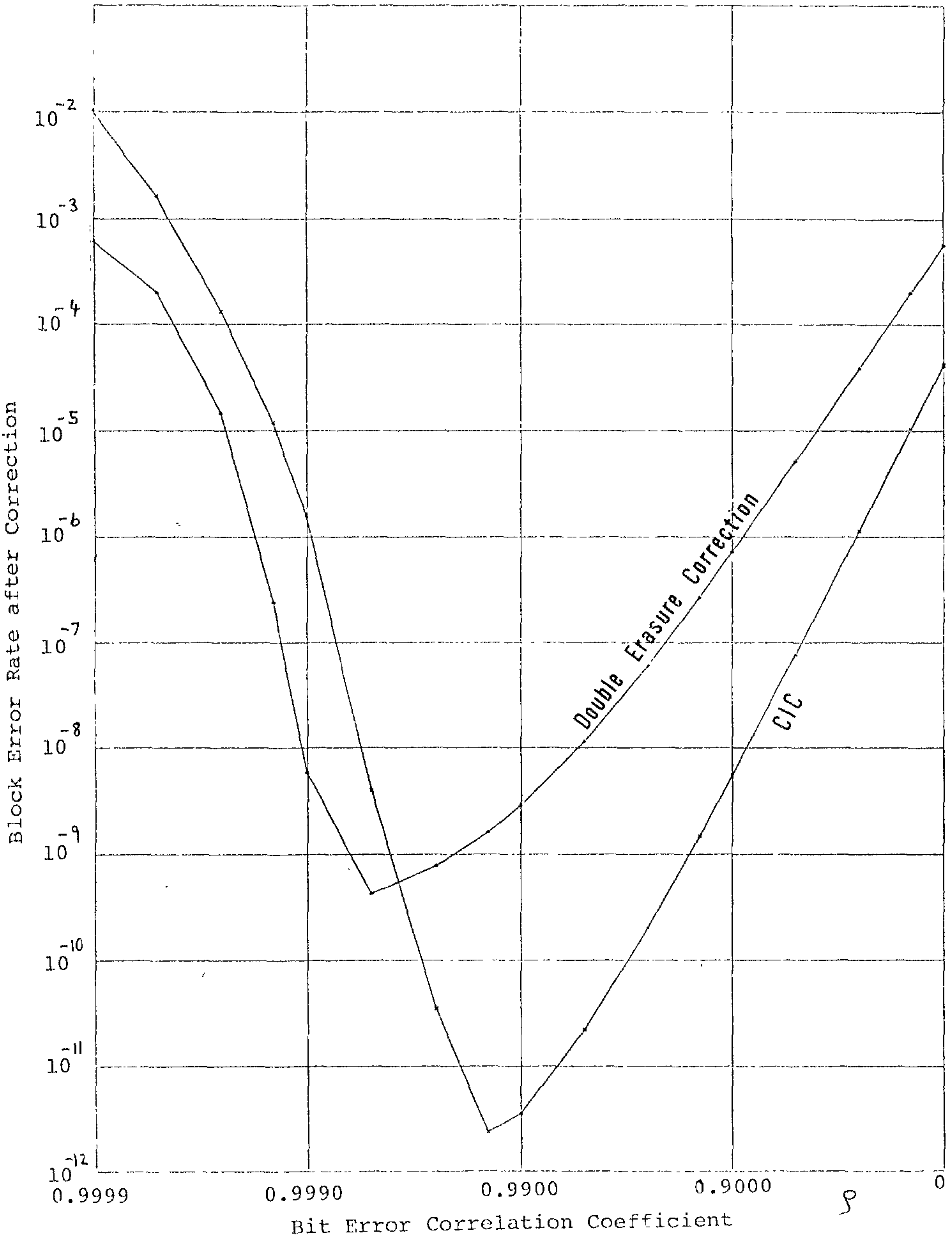


Fig. 12.7 CIC and Double Erasure Correction (5)
k=4, D=24, d=4, Bit Error Rate=10^{-4}

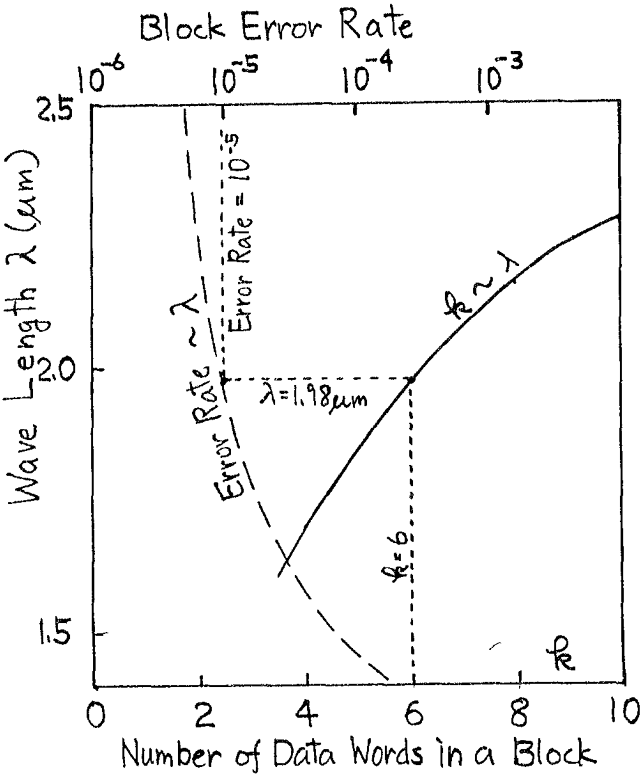
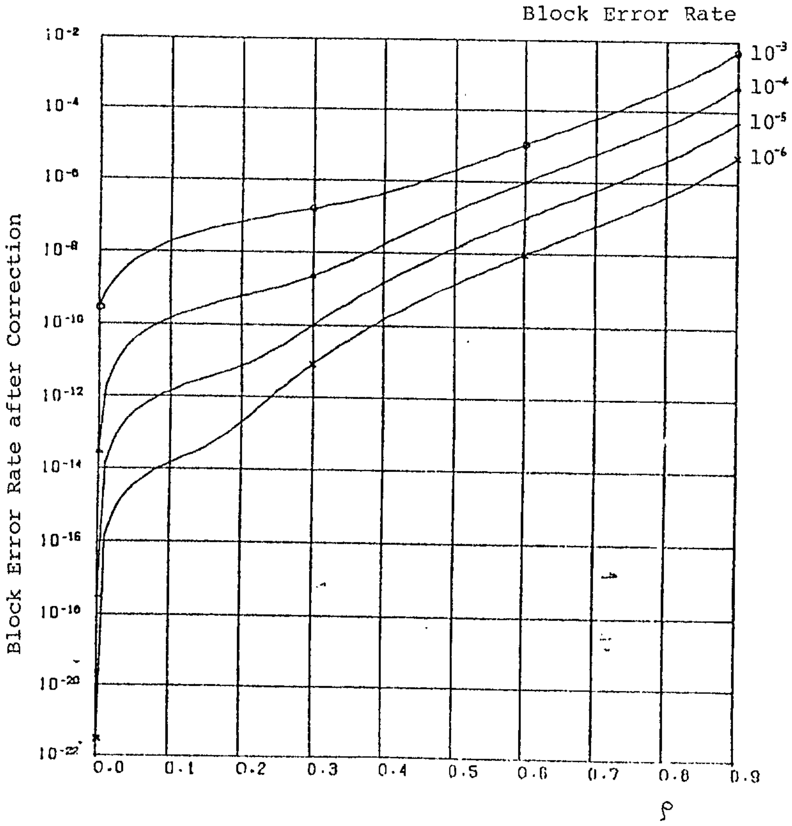


Fig. 12.8 k and Wave Length

Fig. 12.9 Performance of CIC (1)

Decoding Step = 2 (Q-P)
k = 6, D = 8, d = 1



Block Error Correlation Coefficient

Fig. 12.10 Performance of CIC (2)

Decoding Step = 2 (Q→P)
k = 6, D = 8, d = 2

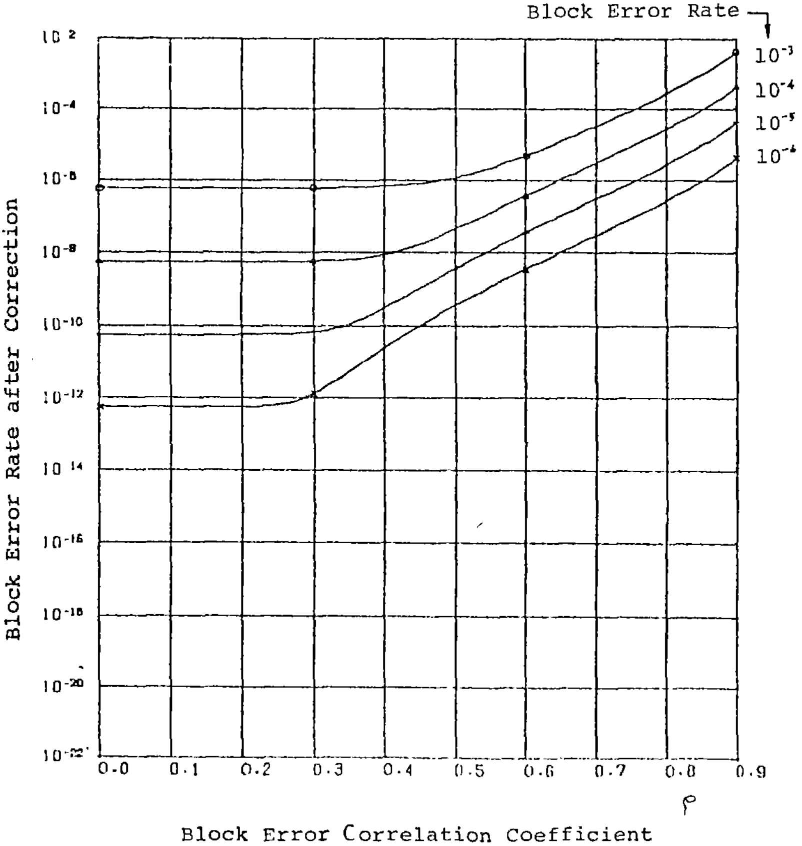


Fig. 12.11 Performance of CIC (3)

Decoding Step = 2 ($\rho \to P$)
k = 6, D = 8, d = 3

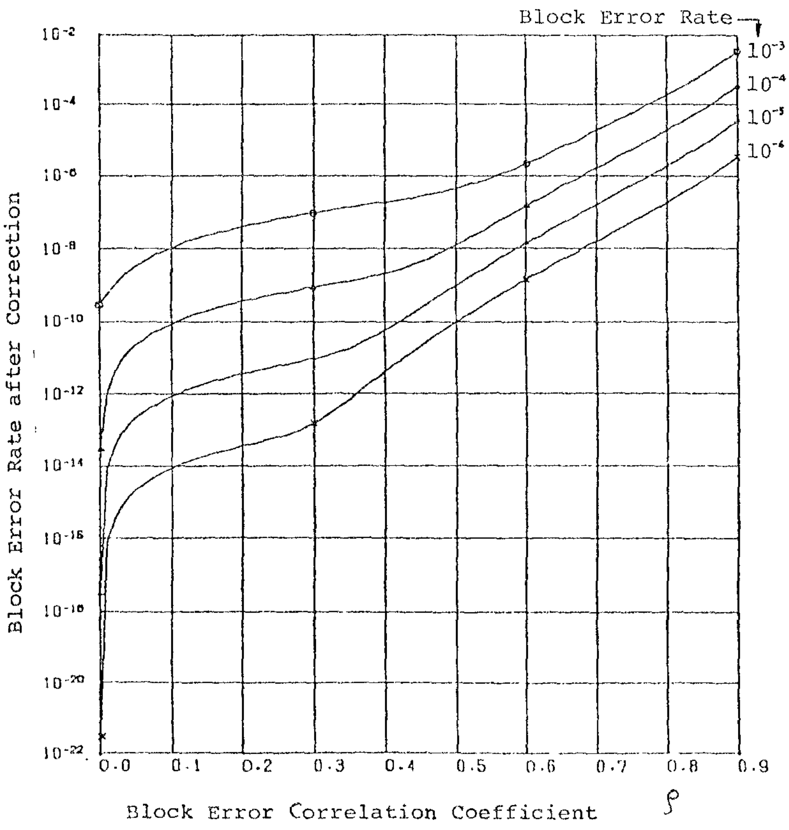


Fig. 12.12 Performance of CIC (4)

Decoding Step = 2 (ρ→P)
k = 6, D = 12, d = 1

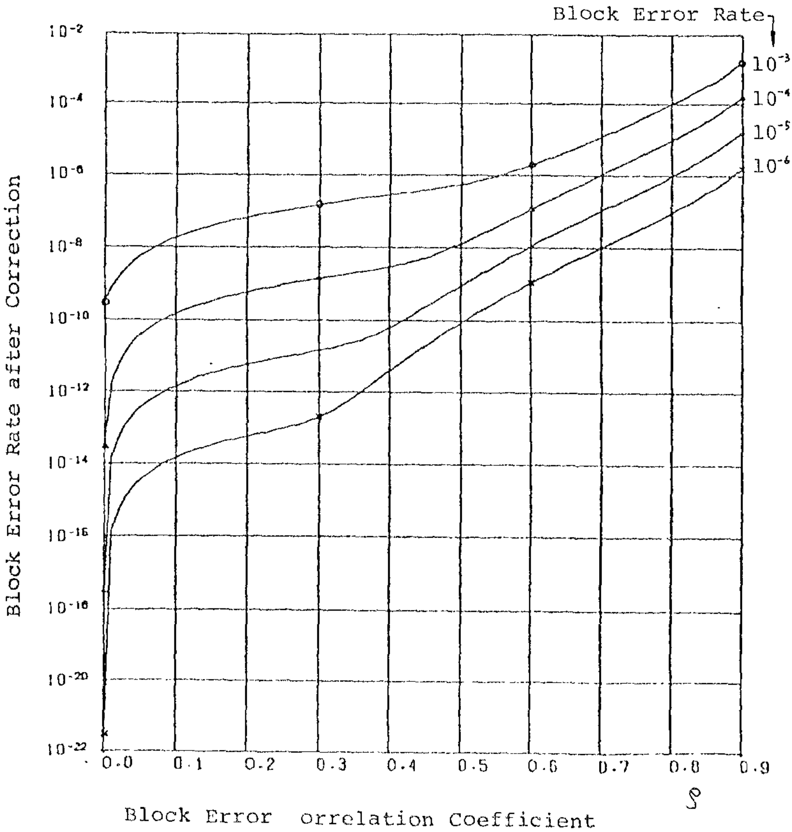


Fig. 12.13 Performance of CIC (5)

Decoding Step = 2 (Q→P)
k = 6, D = 12, d = 2

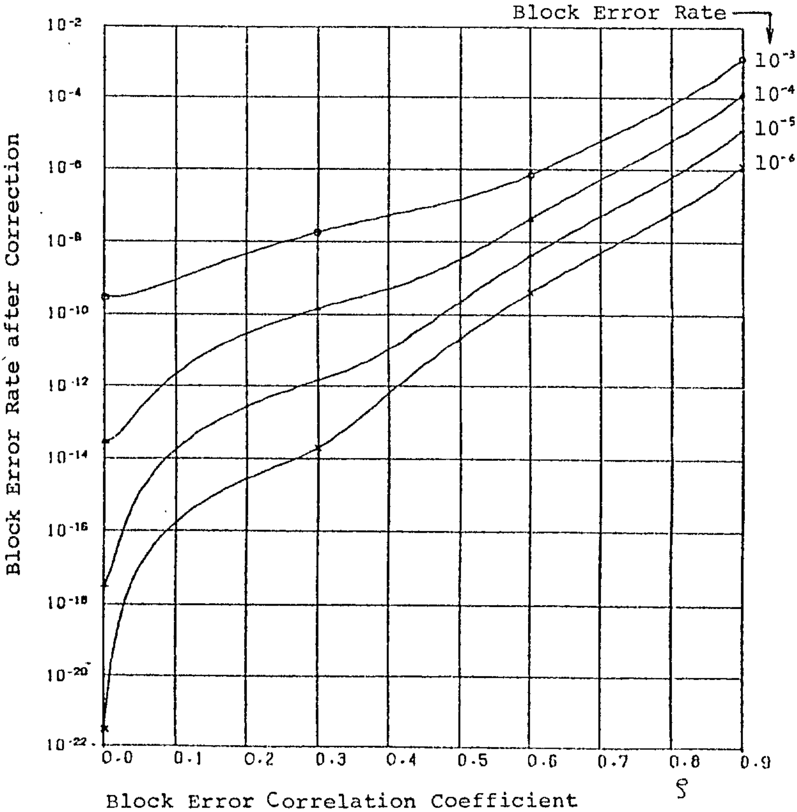


Fig. 12.14 Performance of CIC (6)

Decoding Step = 2 (Q→P)
k = 6, D = 12, d = 3

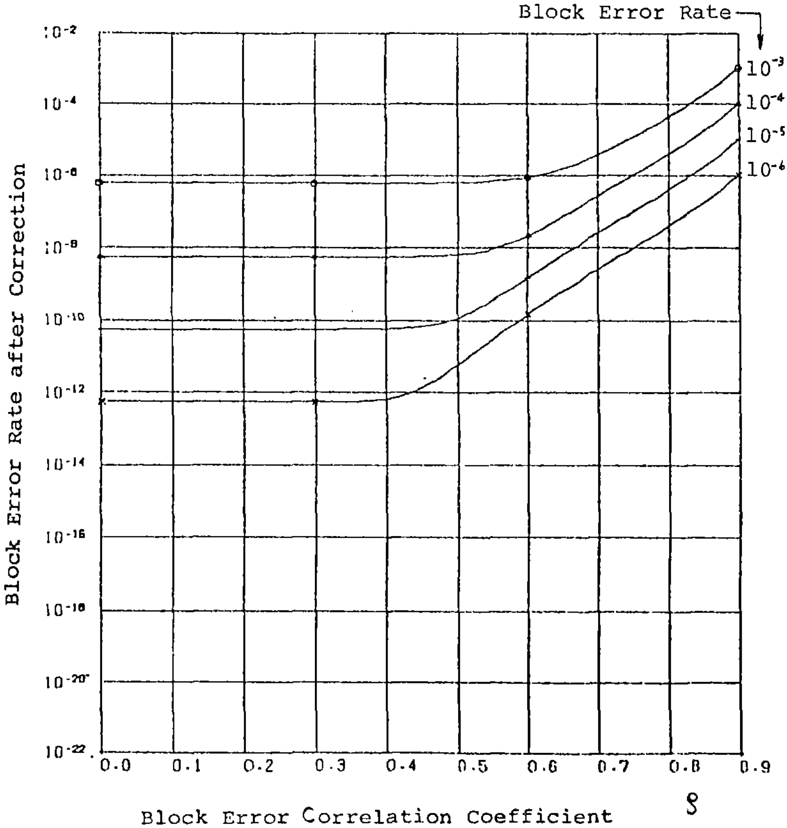


Fig. 12.15 Performance of CIC (7)

Decoding Step = 2 (Q-P)
k = 6, D = 17, d = 1

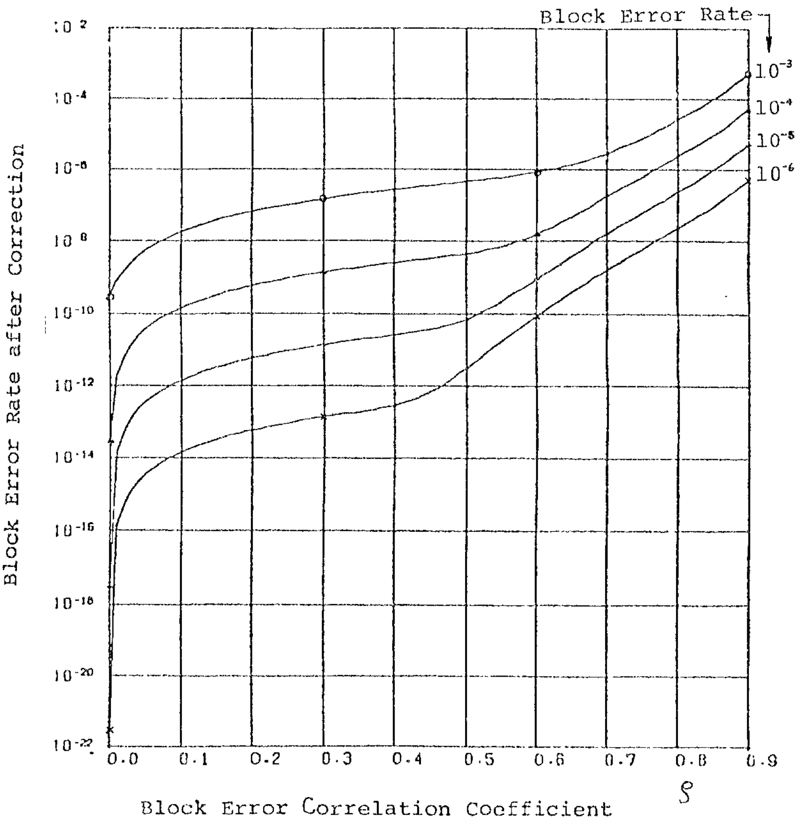


Fig. 12.16 Performance of CIC (8)

Decoding Step = 2 (Q→P)
k = 6, D = 17, d = 2

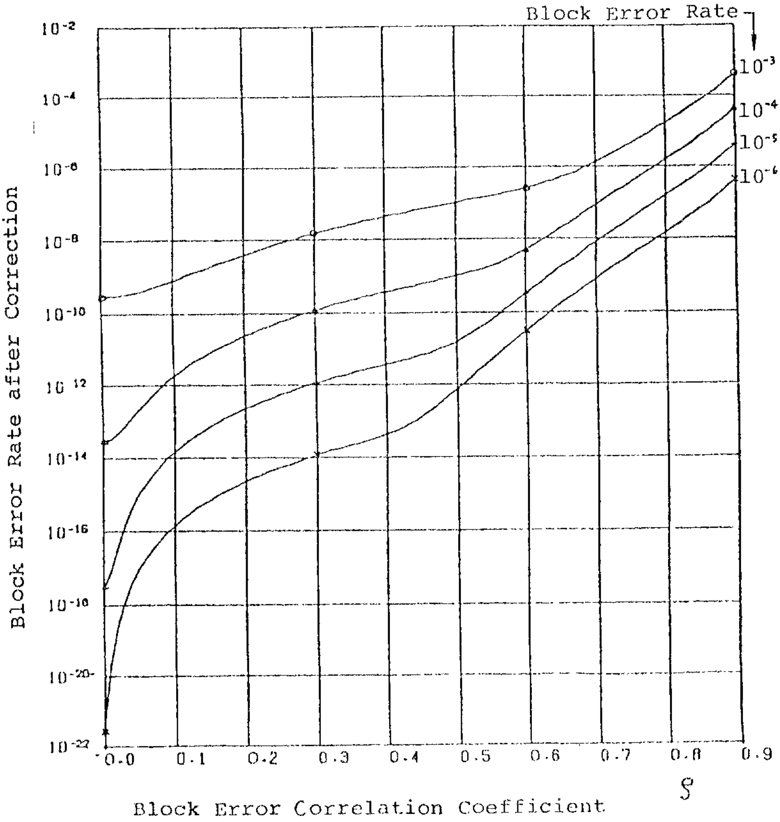


Fig. 12.17 Performance of CIC (9)

Decoding Step = 2 (Q→P)
k = 6, D = 17, d = 3

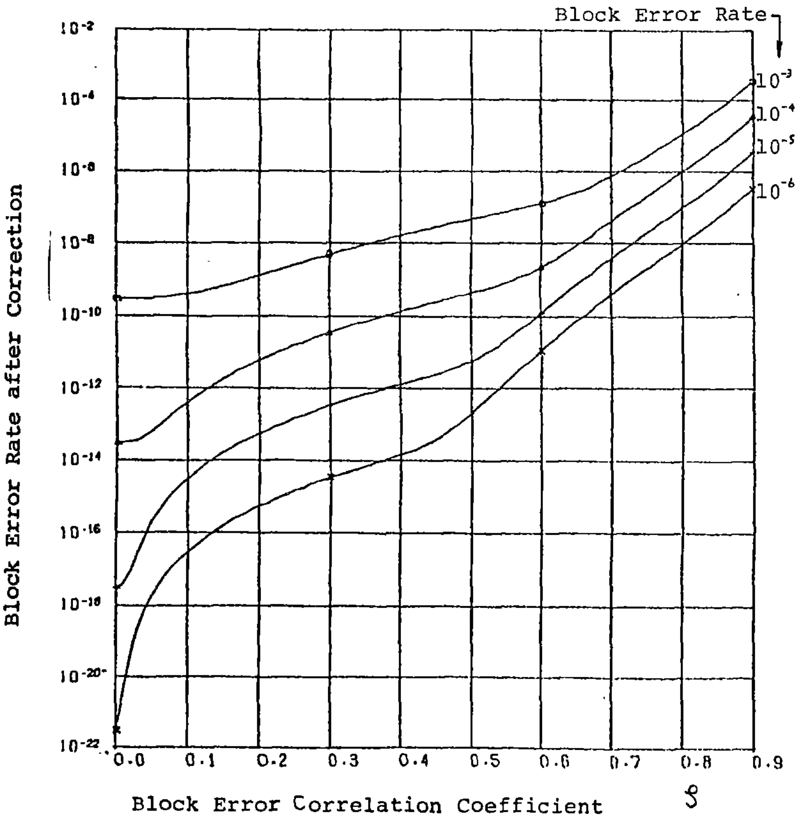


Fig. 12.18 Performance of CIC (10)

Decoding Step = 3 (P-Q-P)
k = 6, D = 8, d = 1

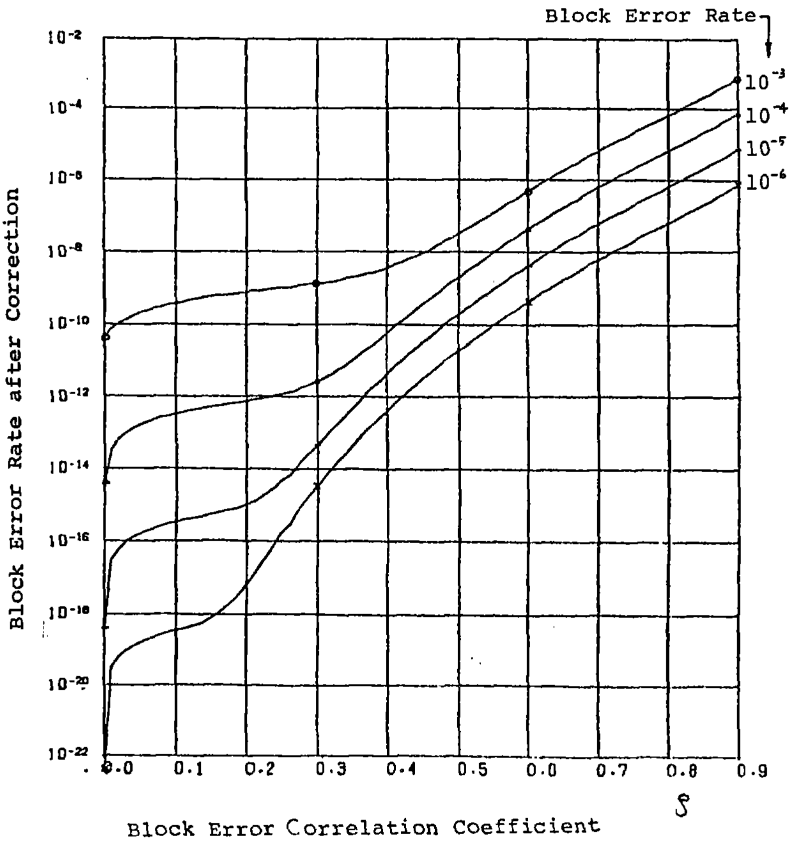


Fig. 12.19 Performance of CIC (11)

Decoding Step = 3 (P-Q-P)
k = 6, D = 8, d = 2

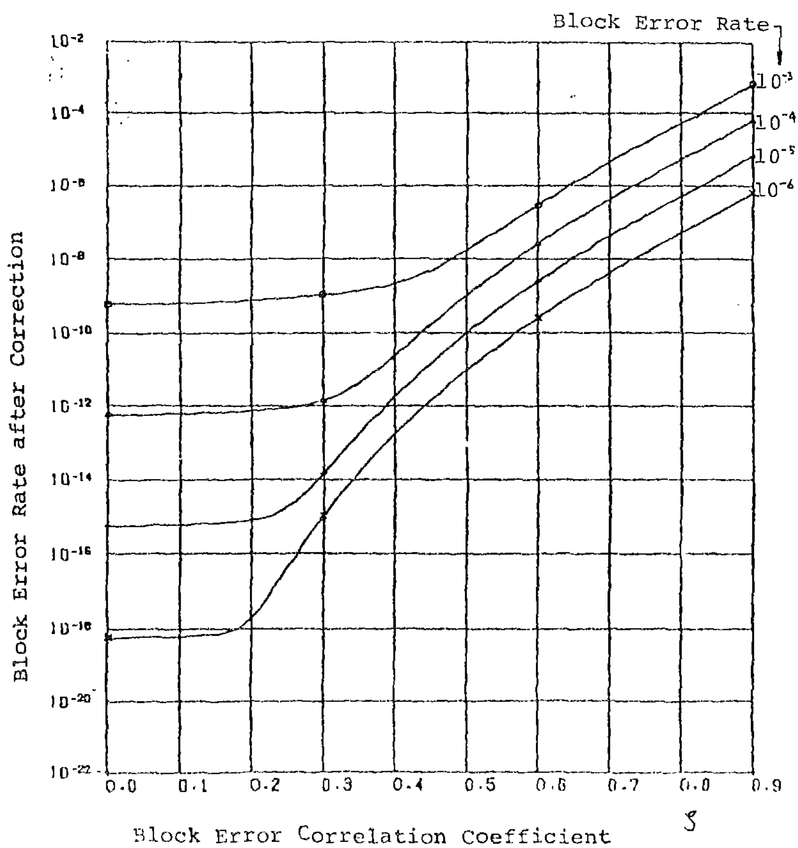


Fig. 12.20 Performance of CIC (12)

Decoding Step = 3(P-Q-P)
k = 6, D = 8, d = 3

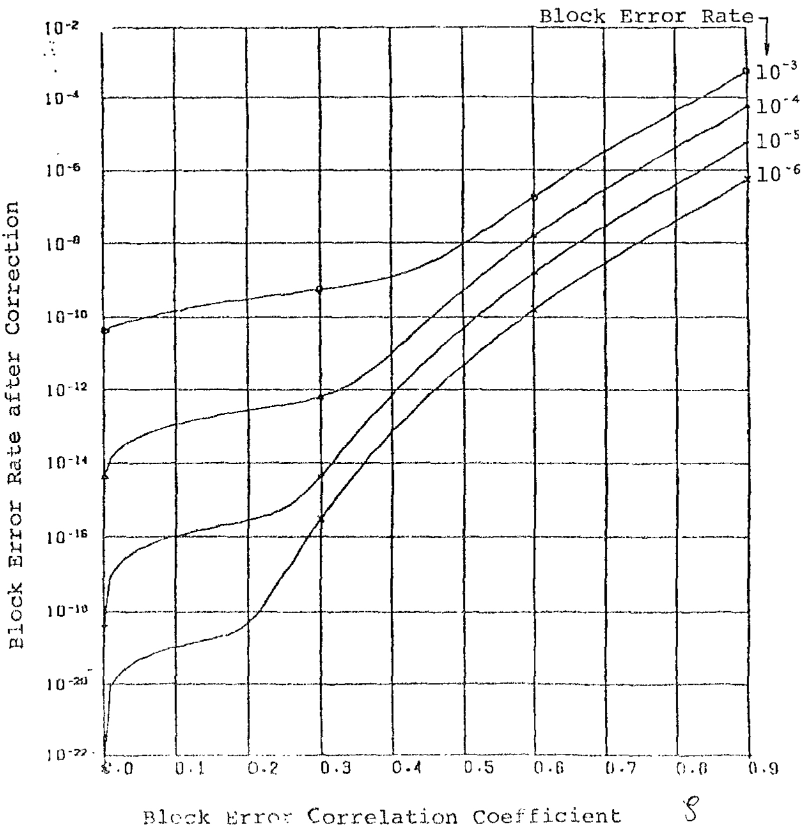


Fig. 12.21 Performance of CIC (13)

Decoding Step = 3 (P-O-P)
k = 6, D = 12, d = 1

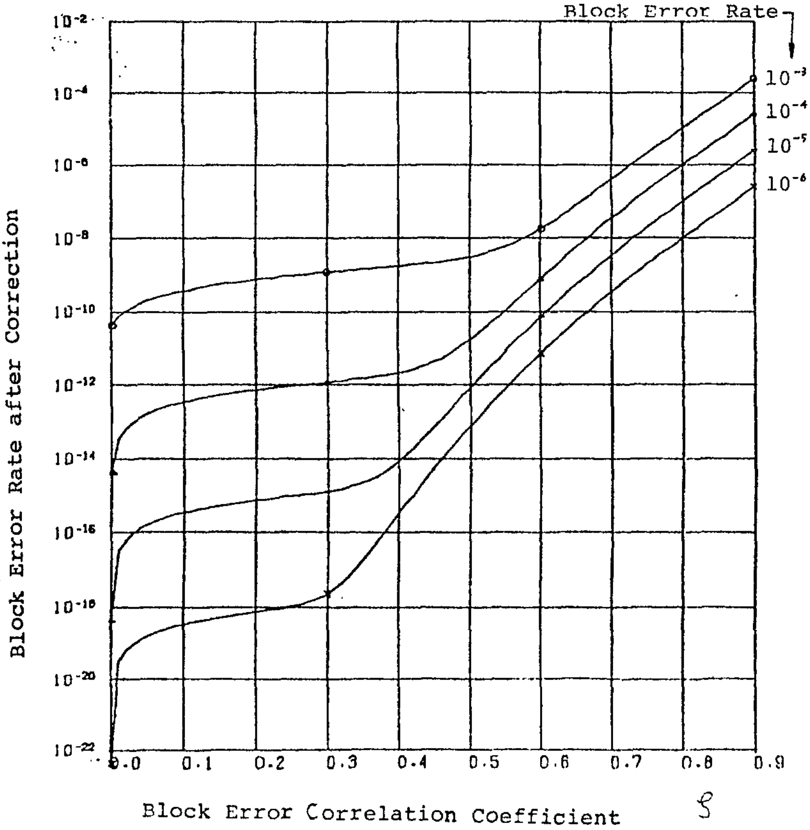


Fig. 12.22 Performance of CIC (14)

Decoding Step = 3 (P-Q-P)
k = 6, D = 12, d = 2

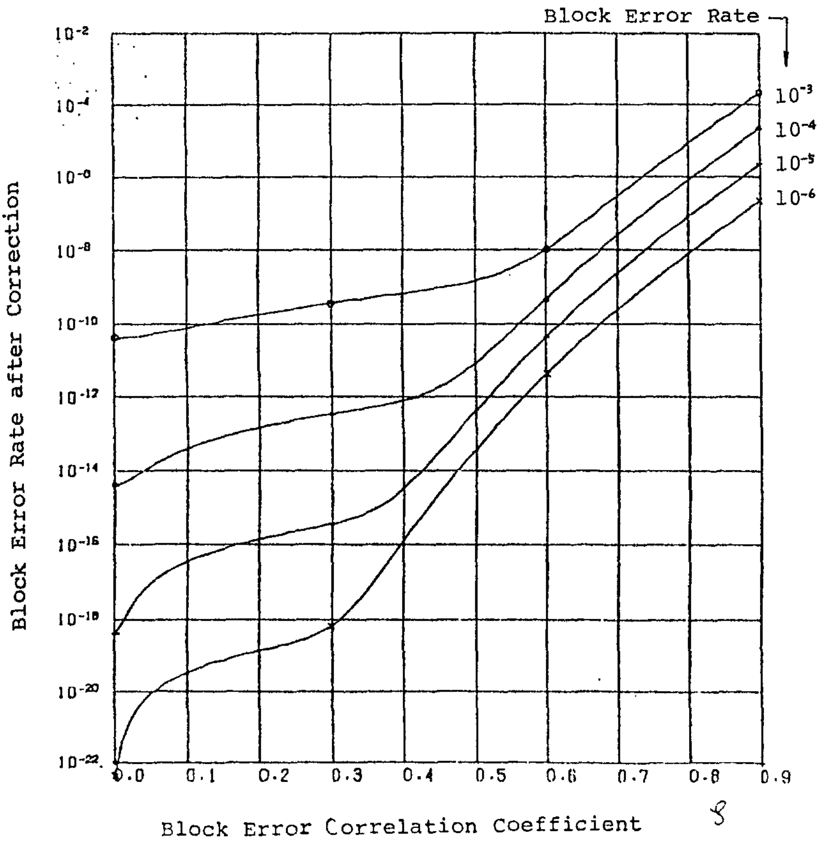


Fig. 12.23 Performance of CIC (15)

Decoding Step = 3 (P-Q-P)
k = 6, D = 12, d = 3

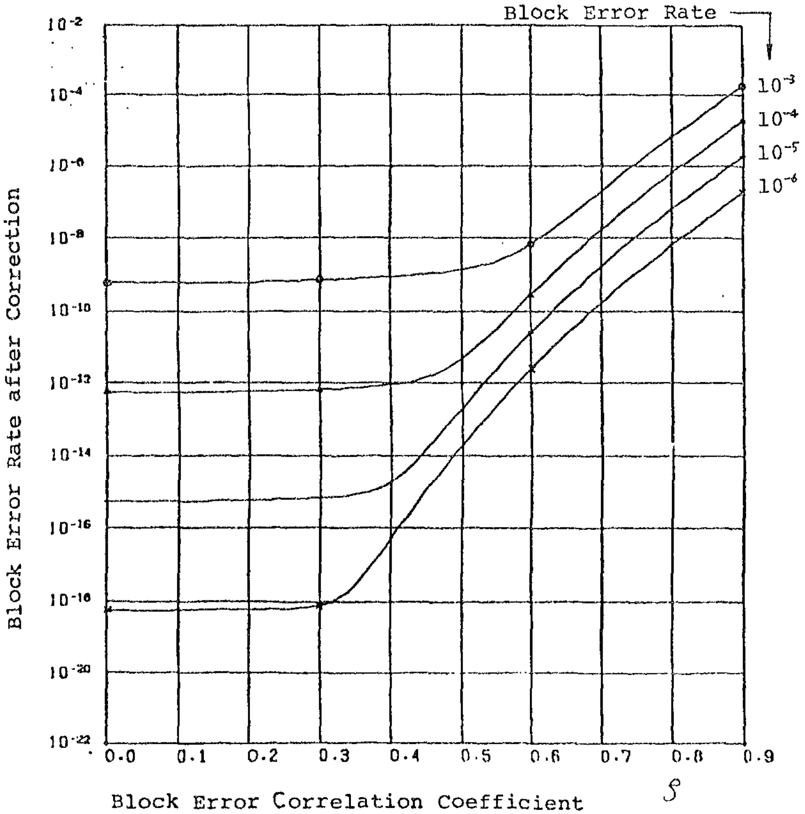


Fig. 12. 24 Performance of CIC (16)

Decoding Step = 3 (P-Q-P)
k = 6, D = 17, d = 1

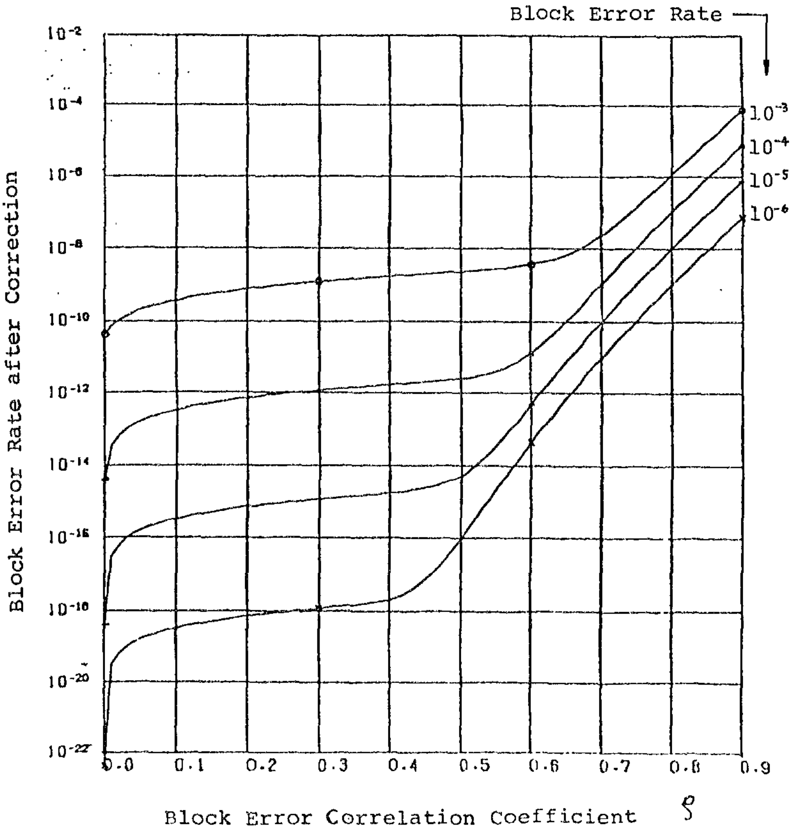


Fig. 12.25 Performance of CIC (17)

Decoding Step = 3 (P-Q-P)
k = 7, D = 17, d = 2

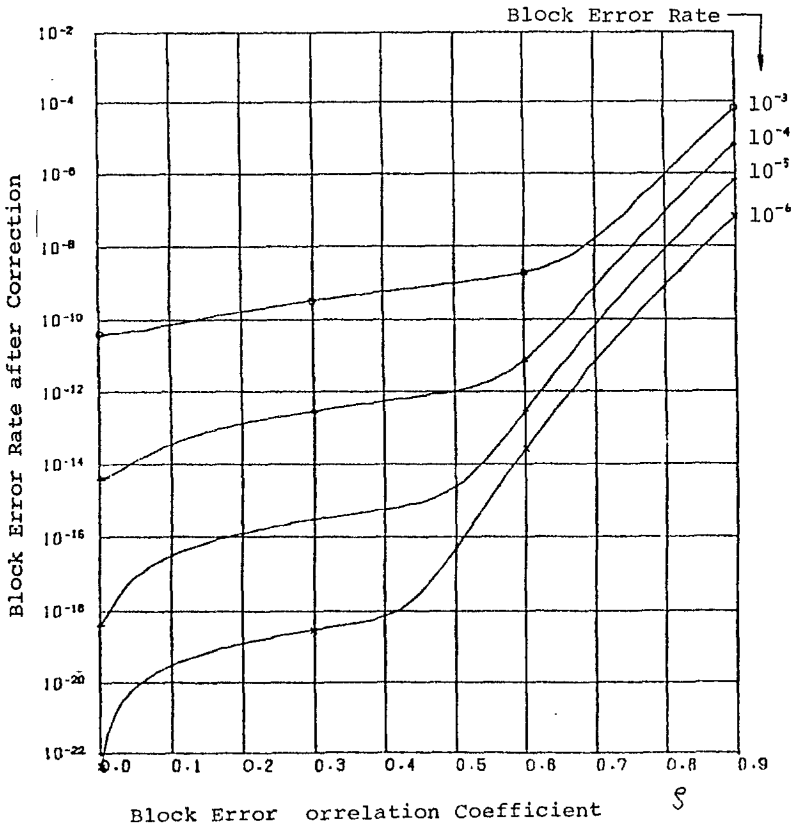
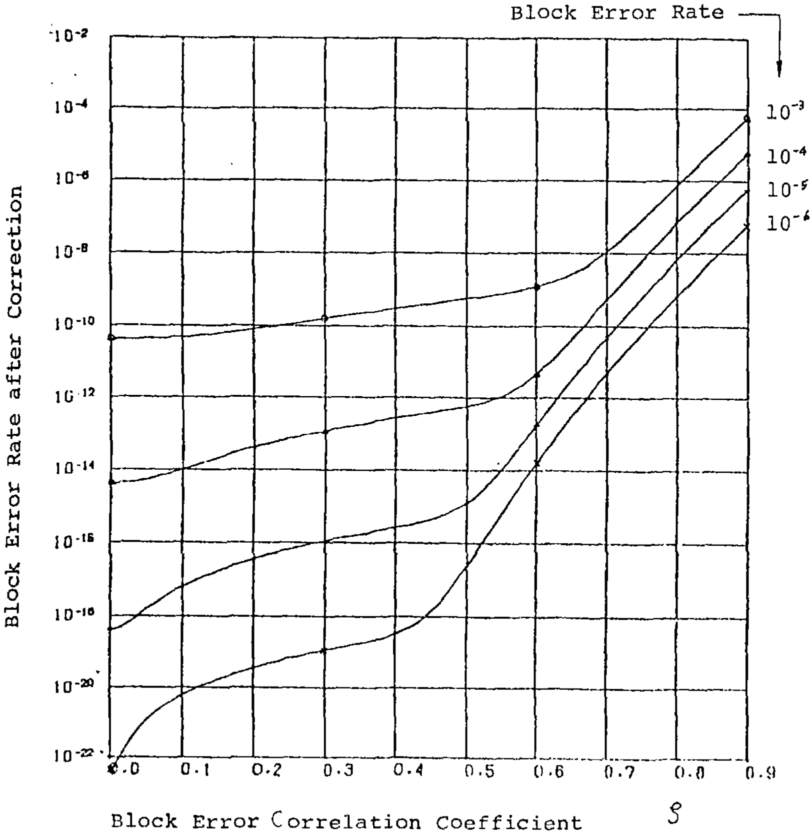


Fig. 12.26 Performance of CIC (18)

Decoding Step = 3 (P-Q-P)
k = 6, k = 17, d = 3



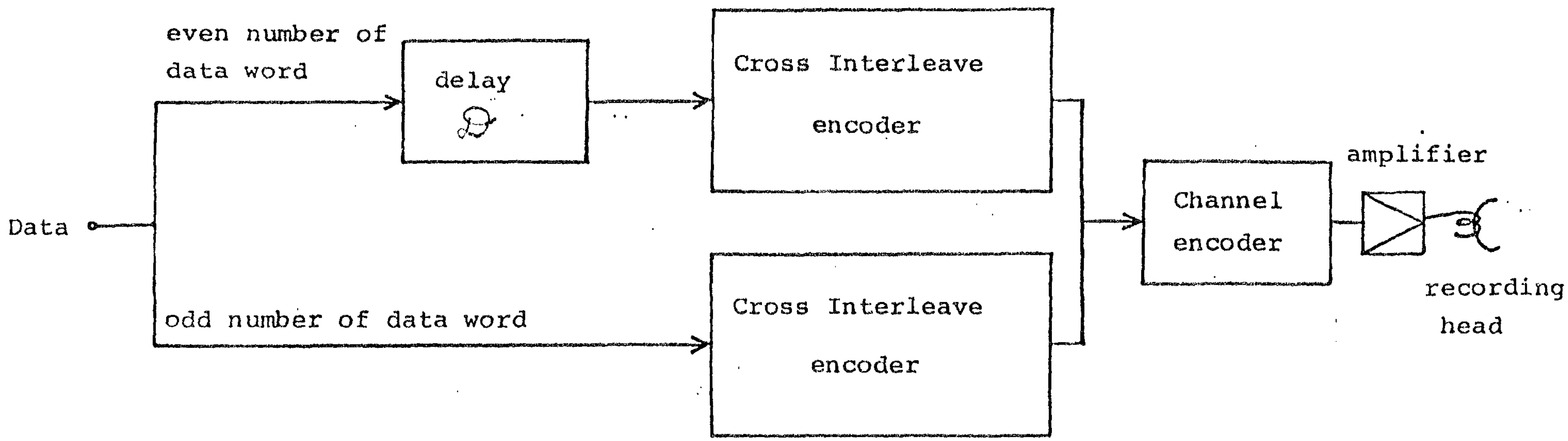


Fig 12.27 Recording system

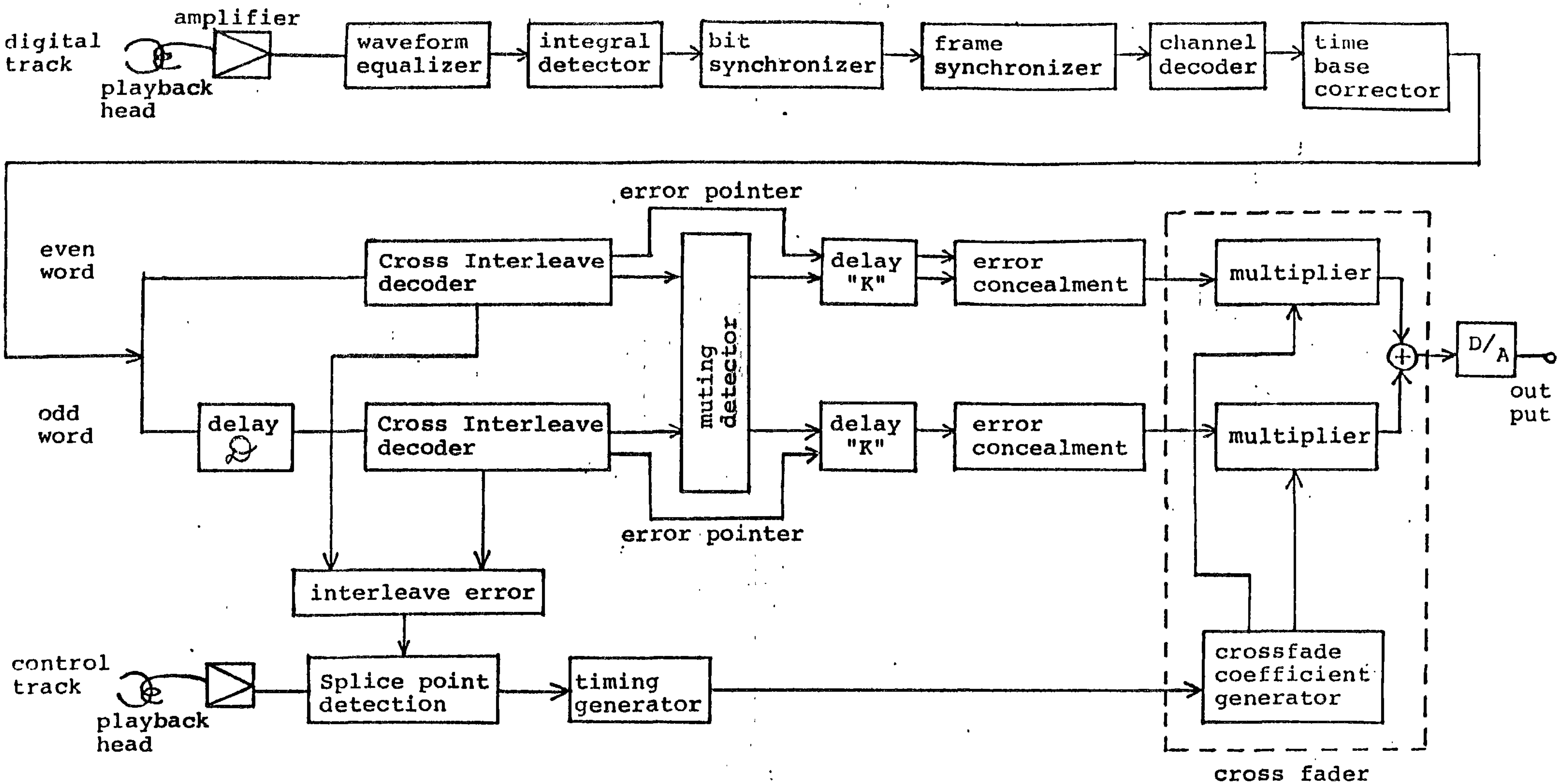


Fig. 12.28 Playing back system

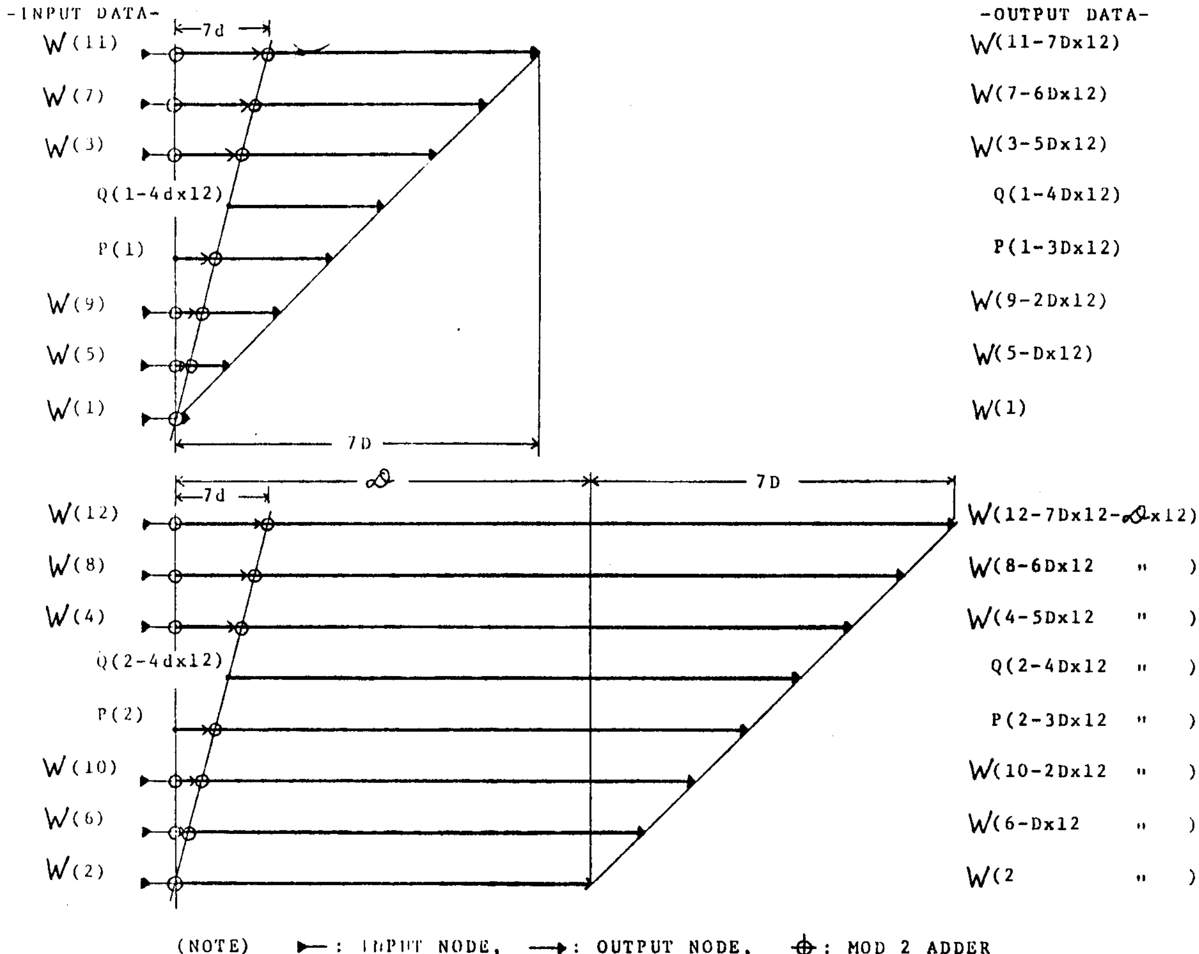
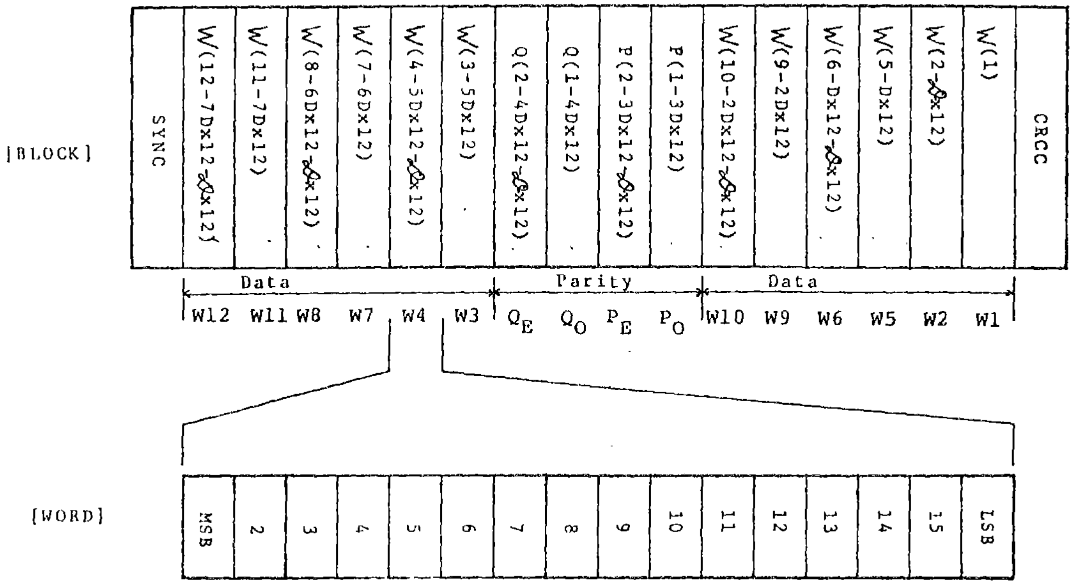


Fig. 12.29 DASH Encoding Scheme

FIG.12.30 THE CONSTRUCTION OF WORD AND BLOCK (DASH)



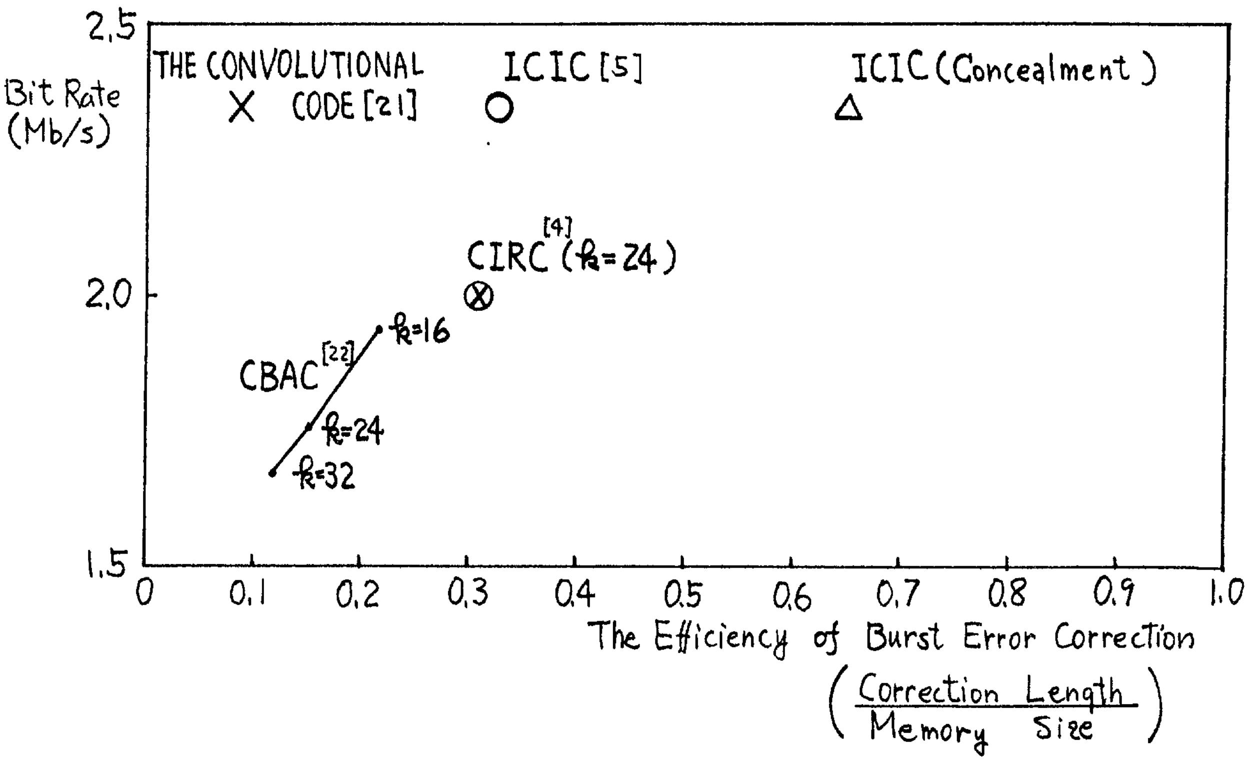
( ) : Word Seqence

dl : Odd-Even Delay (=204 Blocks)

D : Unit Interleave (=17 ")

d : Q-Seqence Unit Delay (=2 ")

Fig. 13.1 Burst Error Correction and Bit Rate



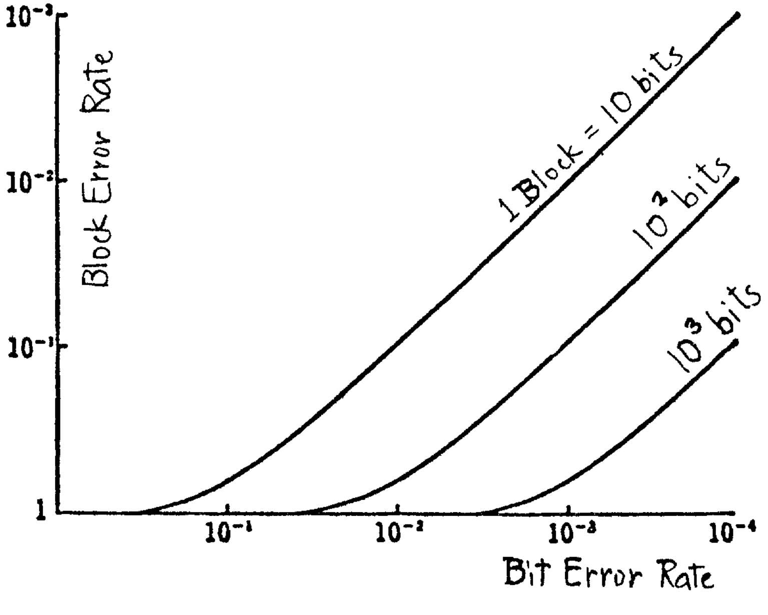


Fig. 13.2 Block Error Rate for Random Bit Error

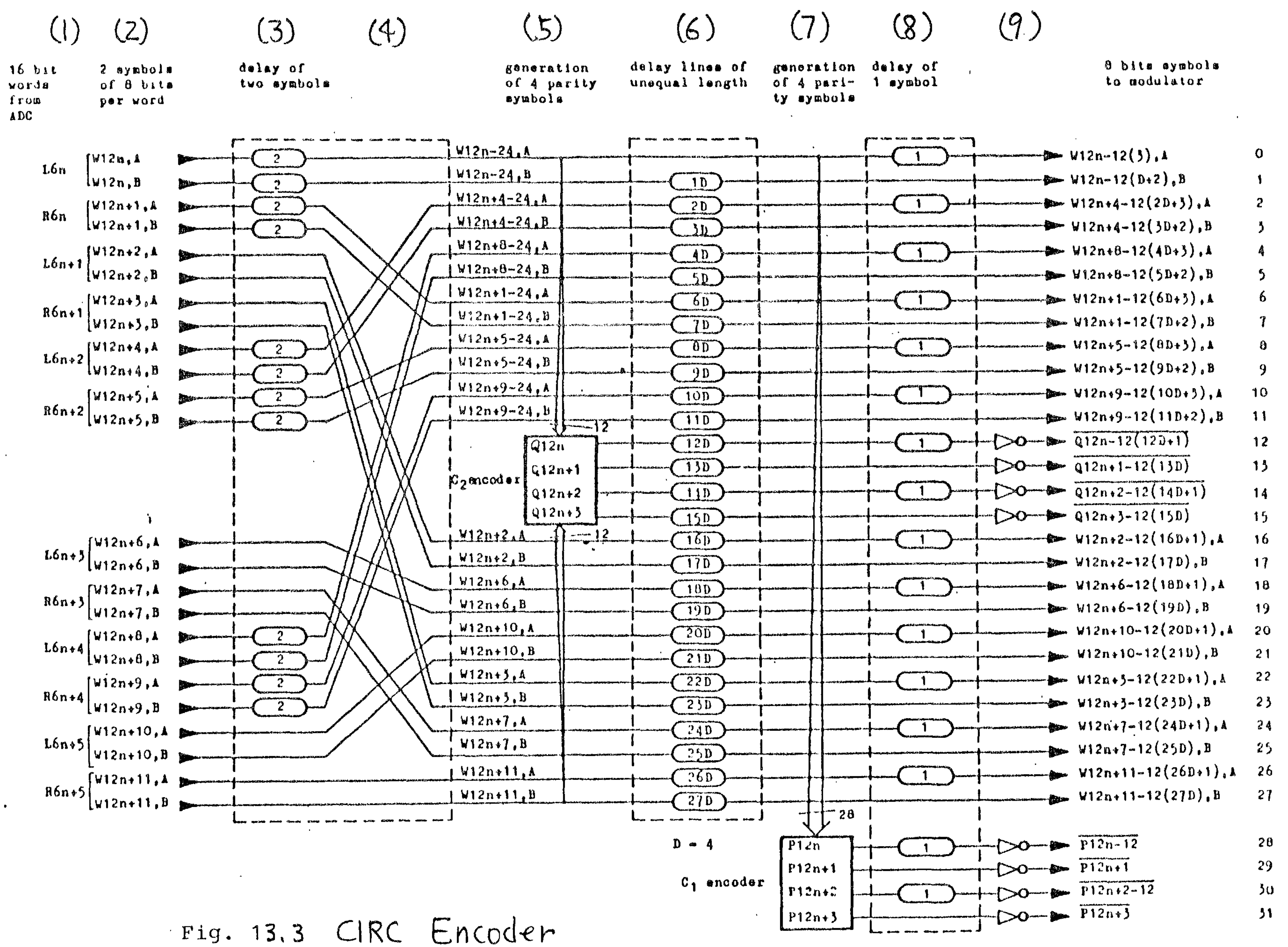


Fig. 13.3 CIRC Encoder

Fig. 13.4 Parity Check Matrix of CIRC

$$H_P = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \alpha^{31} & \alpha^{30} & \alpha^{29} & \alpha^{28} & \alpha^{27} & \alpha^{26} & \alpha^{25} & \alpha^{24} & \alpha^{23} & \alpha^{22} & \alpha^{21} & \alpha^{20} & \alpha^{19} & \alpha^{18} & \alpha^{17} & \alpha^{16} & \alpha^{15} & \alpha^{14} & \alpha^{13} & \alpha^{12} & \alpha^{11} & \alpha^{10} & \alpha^{9} & \alpha^{8} & \alpha^{7} & \alpha^{6} & \alpha^{5} & \alpha^{4} & \alpha^{3} & \alpha^{2} & \alpha^{1} & 1 \\ \alpha^{62} & \alpha^{60} & \alpha^{58} & \alpha^{56} & \alpha^{54} & \alpha^{52} & \alpha^{50} & \alpha^{48} & \alpha^{46} & \alpha^{44} & \alpha^{42} & \alpha^{40} & \alpha^{38} & \alpha^{36} & \alpha^{34} & \alpha^{32} & \alpha^{30} & \alpha^{28} & \alpha^{26} & \alpha^{24} & \alpha^{22} & \alpha^{20} & \alpha^{18} & \alpha^{16} & \alpha^{14} & \alpha^{12} & \alpha^{10} & \alpha^{8} & \alpha^{6} & \alpha^{4} & \alpha^{2} & 1 \\ \alpha^{93} & \alpha^{90} & \alpha^{87} & \alpha^{84} & \alpha^{81} & \alpha^{78} & \alpha^{75} & \alpha^{72} & \alpha^{69} & \alpha^{66} & \alpha^{63} & \alpha^{60} & \alpha^{57} & \alpha^{54} & \alpha^{51} & \alpha^{48} & \alpha^{45} & \alpha^{42} & \alpha^{39} & \alpha^{36} & \alpha^{33} & \alpha^{30} & \alpha^{27} & \alpha^{24} & \alpha^{21} & \alpha^{18} & \alpha^{15} & \alpha^{12} & \alpha^{9} & \alpha^{6} & \alpha^{3} & 1 \end{bmatrix}$$

$$H_Q = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \alpha^{27} & \alpha^{26} & \alpha^{25} & \alpha^{24} & \alpha^{23} & \alpha^{22} & \alpha^{21} & \alpha^{20} & \alpha^{19} & \alpha^{18} & \alpha^{17} & \alpha^{16} & \alpha^{15} & \alpha^{14} & \alpha^{13} & \alpha^{12} & \alpha^{11} & \alpha^{10} & \alpha^{9} & \alpha^{8} & \alpha^{7} & \alpha^{6} & \alpha^{5} & \alpha^{4} & \alpha^{3} & \alpha^{2} & \alpha^{1} & 1 \\ \alpha^{54} & \alpha^{52} & \alpha^{50} & \alpha^{48} & \alpha^{46} & \alpha^{44} & \alpha^{42} & \alpha^{40} & \alpha^{38} & \alpha^{36} & \alpha^{34} & \alpha^{32} & \alpha^{30} & \alpha^{28} & \alpha^{26} & \alpha^{24} & \alpha^{22} & \alpha^{20} & \alpha^{18} & \alpha^{16} & \alpha^{14} & \alpha^{12} & \alpha^{10} & \alpha^{8} & \alpha^{6} & \alpha^{4} & \alpha^{2} & 1 \\ \alpha^{81} & \alpha^{78} & \alpha^{75} & \alpha^{72} & \alpha^{69} & \alpha^{66} & \alpha^{63} & \alpha^{60} & \alpha^{57} & \alpha^{54} & \alpha^{51} & \alpha^{48} & \alpha^{45} & \alpha^{42} & \alpha^{39} & \alpha^{36} & \alpha^{33} & \alpha^{30} & \alpha^{27} & \alpha^{24} & \alpha^{21} & \alpha^{18} & \alpha^{15} & \alpha^{12} & \alpha^{9} & \alpha^{6} & \alpha^{3} & 1 \end{bmatrix}$$

Fig. 13.5 Data Vector of CIRC

$$
V_p = \begin{pmatrix}
W12n-12(2),A \\
W12n-12(1D+2),B \\
W12n+4-12(2D+2),A \\
W12n+4-12(3D+2),B \\
W12n+8-12(4D+2),A \\
W12n+8-12(5D+2),B \\
W12n+1-12(6D+2),A \\
W12n+1-12(7D+2),B \\
W12n+5-12(8D+2),A \\
W12n+5-12(9D+2),B \\
W12n+9-12(10D+2),A \\
W12n+9-12(11D+2),B \\
Q12n-12(12D) \\
Q12n-1-12(13D) \\
Q12n+2-12(14D) \\
Q12n+3-12(15D) \\
W12n+2-12(16D),A \\
W12n-2-12(17D),B \\
W12n+6-12(18D),A \\
W12n+6-12(19D),B \\
W12n+10-12(20D),A \\
W12n+10-12(21D),B \\
W12n+3-12(22D),A \\
W12n+3-12(23D),B \\
W12n+7-12(24D),A \\
W12n+7-12(25D),B \\
W12n+11-12(26D),A \\
W12n+11-12(27D),B \\
P12n \\
P12n+1 \\
P12n+2 \\
P12n-3
\end{pmatrix}
\qquad
V_q = \begin{pmatrix}
W12n-24,A \\
W12n-24,B \\
W12n+4-24,A \\
W12n+4-24,B \\
W12n+8-24,A \\
W12n+8-24,B \\
W12n+1-24,A \\
W12n+1-24,B \\
W12n+5-24,A \\
W12n+5-24,B \\
W12n+9-24,A \\
W12n+9-24,B \\
Q12n \\
Q12n+1 \\
Q12n-2 \\
Q12n+3 \\
W12n+2,A \\
W12n+2,B \\
W12n+6,A \\
W12n+6,B \\
W12n+10,A \\
W12n+10,B \\
W12n+3,A \\
W12n+3,B \\
W12n+7,A \\
W12n+7,B \\
W12n+11,A \\
W12n+11,B
\end{pmatrix}
$$

$D = 4$

$n = 0, 1, 2,$

Fig. 13.6 CIRC Decoder

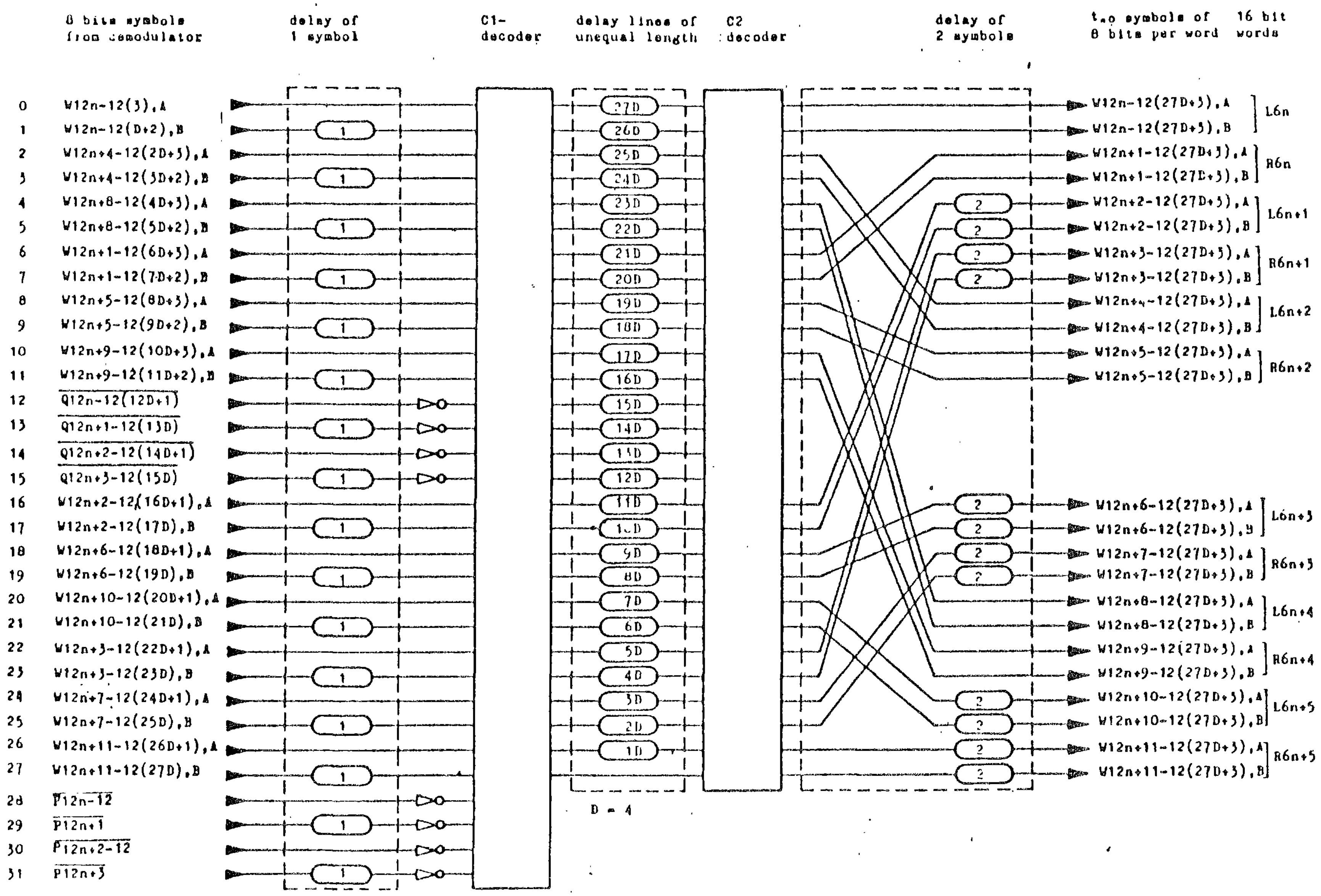
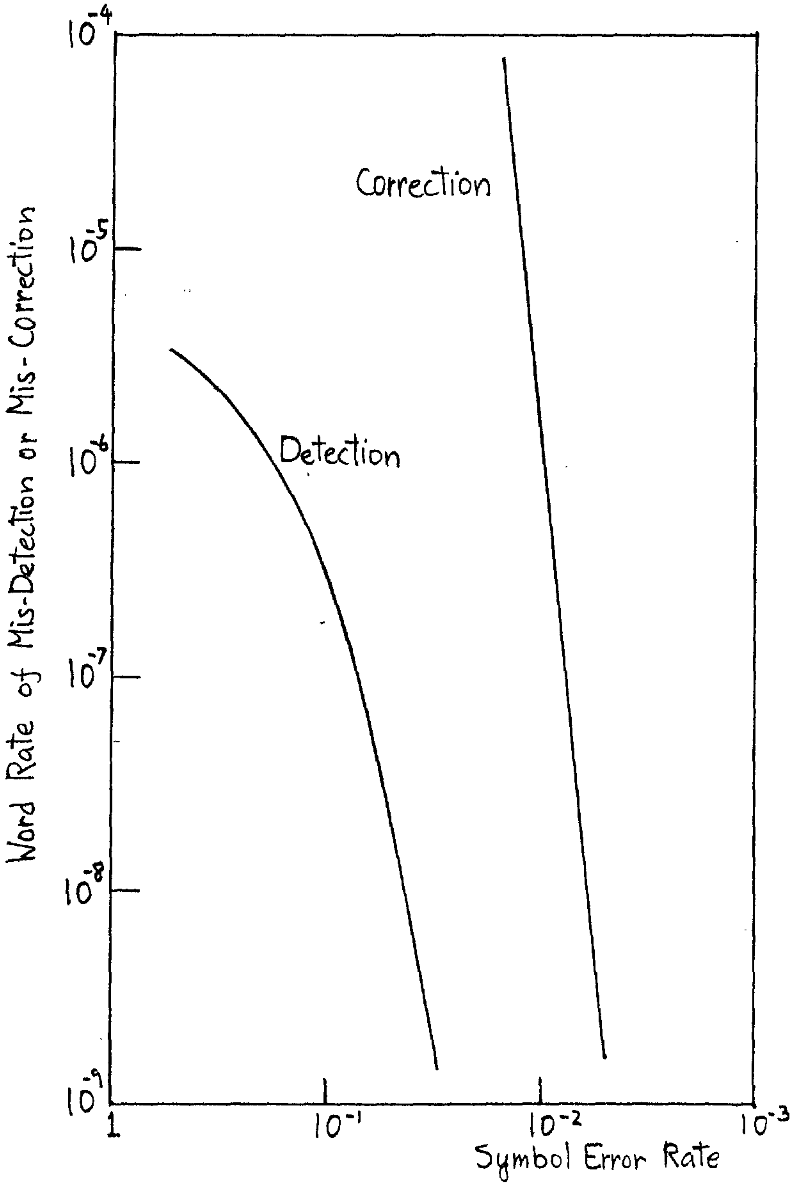


Fig. 13.7 Performance of CIRC (Random Symbol Error)



		a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
DATA WORDS	A	1	0	1	1	0	1	1	1
	B	1	0	0	1	1	0	1	1
	C	0	1	0	1	0	0	1	0
	D	0	1	1	0	1	0	1	0
CHECK WORDS	P	0	0	0	1	0	1	0	0
	Q	1	0	1	0	0	0	0	0

CHECK WORDS

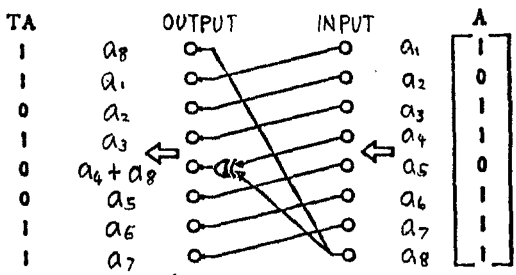
$P = A + B + C + D$

$Q = T^4A + T^3B + T^2C + TD$

Matrix T

Generator Polynomial $G(x) = x^7 + x^3$

$$\underbrace{\begin{bmatrix} a_8 \\ a_1 \\ a_2 \\ a_3 \\ a_4 + a_8 \\ a_5 \\ a_6 \\ a_7 \end{bmatrix}}_{TA} = \underbrace{\begin{bmatrix} 0&0&0&0&0&0&0&1 \\ 1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&1 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&1&0 \end{bmatrix}}_{T} \underbrace{\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{bmatrix}}_{A}$$



(Circuit of Matrix T)

A	1 0 1 1 0 1 1 1
TA	1 1 0 1 0 0 1 1
T^2A	1 1 1 0 0 0 0 1
T^3A	1 1 1 1 1 0 0 0
T^4A	0 1 1 1 1 1 0 0

B	1 0 0 1 1 0 1 1
TB	1 1 0 0 0 1 0 1
T^2B	1 1 1 0 1 0 1 0
T^3B	0 1 1 1 0 1 0 1

C	0 1 0 1 0 0 1 0
TC	0 0 1 0 1 0 0 1
T^2C	1 0 0 1 1 1 0 0

D	0 1 1 0 1 0 1 0
TD	0 0 1 1 0 1 0 1

Fig. A1 An Example of b-Adjacent Code

$$\text{DATA WORDS}\quad \begin{cases} A = (001) \\ B = (101) \\ C = (011) \\ D = (100) \end{cases}$$

$$\text{CHECK WORDS}\quad \begin{cases} P = (111) \\ Q = (110) \end{cases}$$

Check words P and Q are the solutions of the following equations.

$$\begin{cases} A + B + C + D + P + Q = 0 \\ \alpha^6 A + \alpha^5 B + \alpha^4 C + \alpha^3 D + \alpha^2 P + \alpha Q = 0 \end{cases}$$

$$\therefore \begin{cases} P = \alpha A + \alpha^2 B + \alpha^5 C + \alpha^3 D \\ Q = \alpha^3 A + \alpha^6 B + \alpha^4 C + \alpha D \end{cases}$$

$$\text{where:} \begin{cases} \alpha = (010) & \alpha^5 = (111) \\ \alpha^2 = (100) & \alpha^6 = (101) \\ \alpha^3 = (011) & \alpha^7 = (001) = 1 \\ \alpha^4 = (110) & 0 = (000) \end{cases}$$

Syndromes;

$$\begin{cases} S = A' + B' + C' + D' + P' + Q' \\ S = \alpha^6 A' + \alpha^5 B' + \alpha^4 C' + \alpha^3 D' + \alpha^2 P' + \alpha Q' \end{cases}$$

Fig. B1 An Example of Reed Solomon Code

bits		000	001	010	011	100	101	110	111
	elements	0	1	α	α^3	α^2	α^6	α^4	α^5
000	0	0	0	0	0	0	0	0	0
001	1	0	1	α	α^3	α^2	α^6	α^4	α^5
010	α	0	α	α^2	α^4	α^3	1	α^5	α^6
011	α^3	0	α^3	α^4	α^6	α^5	α^2	1	α
100	α^2	0	α^2	α^3	α^5	α^4	α	α^6	1
101	α^6	0	α^6	1	α^2	α	α^5	α^3	α^4
110	α^4	0	α^4	α^5	1	α^6	α^3	α	α^2
111	α^5	0	α^5	α^6	α	1	α^4	α^2	α^3

Fig. B2 Product Law of Three-Bit Word

$\begin{cases} \text{Galois Field } GF(2^3) \\ \text{Modulus } F(x) = x^3 + x + 1 \\ \text{Primitive Element } \alpha = (010) \end{cases}$