A STUDY OF SYNCHRONIZATION TECHNIQUES FOR BINARY CYCLIC CODES

Stafford Tavares, B.Eng., M.S.

A STUDY OF SYNCHRONIZATION TECHNIQUES FOR BINARY CYCLIC CODES

by

Stafford Tavares, B.Eng., M.S.

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy.

Department of Electrical Engineering
McGill University
Montreal, Canada.

July, 1968.

ABSTRACT

The problem of synchronization error, or slip, for binary cyclic codes is examined for both noiseless and noisy channels. The vector-matrix and polynomial representations of cyclic codes are both utilized. To enable cyclic codes to recover synchronism, two different techniques are considered. The first is to form a suitable coset code, and the second is to generate a "subset code" by restraining some of the information symbols. Neither technique alters the length of the code words.

Several new theorems are presented on the ability of coset codes to detect and correct slip. In particular, a class of coset codes is described which can correct both slip and additive error, even when they occur simultaneously. Further, the performance of coset codes of Fire codes in the presence of slip and burst errors is examined. Results are also presented on the synchronization recovery capability of the subset codes.

ACKNOWLEDGEMENTS

The author wishes to express his appreciation to Professor M. Fukada for his guidance and encouragement throughout this research. Thanks are due to Mrs. P. Hyland and Miss M. Douglas for their excellent typing. The financial support of the National Research Council is gratefully acknowledged.

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

1.1 Introduction To Coding

Since the publication of Shannon's fundamental results in Information Theory[49] in 1948, an extensive literature on the design of codes has developed (see Bibliography by Peterson[39,40] and Peterson and Massey[41]). Shannon demonstrated the existence of codes which transmit (in the limit as the length of the code goes to infinity) information at rates arbitrarily close to channel capacity with arbitrarily small error rates. However, his proof was an existence proof and no synthesis procedure to construct his "random codes" was given. These random codes are impractical and much effort has been devoted to finding codes that are easy to encode and decode and have reasonably good information transmission rates as well.

Codes may be classified into two broad classes :

(a) codes which have fixed word length (block codes) and

(b) codes which have variable word length.

The study of block codes has led to codes which have considerable mathematical structure and hence are attractive for practical applications.

A significant development in the study of block codes was the introduction of group codes (also called parity check codes) by Slepian[50,51]. These codes were a generalization of the error correcting codes of Hamming[29] and all "systematic codes".[29] Slepian pointed out that group codes possess the following features of practical interest :

1. all code words are treated alike in transmission ;
2. the encoding is simple to instrument ;
3. maximum likelihood detection is relatively simple to instrument ; and
4. in certain practical cases there exist no better alphabets.[50]

The code words of group codes were shown to correspond to the elements of a suitably defined group.[4,9,39] Further, Elias[14] has shown that for the binary symmetric channel, there exist group codes which transmit at a rate arbitrarily close to channel capacity with an arbitrarily small probability of error. However, like Shannon, Elias only gives an existence proof.

A smaller class of codes than the group codes are the linear codes *. The code words of a linear code define a subspace of a suitably defined vector space. Linear codes are a subset of the group codes, and a group code is a linear code if the number of symbols (or states) in the code is a prime number.[2,39]

Even more structure was introduced by the discovery of a subset of the linear codes called cyclic codes.[43,39] Cyclic codes are characterized by the additional property that any cyclic permutation of the digits of a code word is also a code word. Cyclic codes are reasonably practical as they can be encoded and decoded by relatively simple devices called feedback shift registers[39]. Additional properties of cyclic codes are given in Section 1.4 and at other appropriate places in the text.

---

* Linear codes can be encoded and decoded by means of linear finite-state switching circuits.[39]

1.2 Outline of Problem

Consider the transmission of digital information over a discrete noisy channel, and assume that the information digits are encoded in a block code. The channel noise causes substitution errors and hence the received symbols may differ from the transmitted symbols. Usually, the reason for the encoding is to protect the information digits from the substitution errors (or additive errors). To achieve this, the encoder adds redundant symbols (or parity check symbols) to the information symbols to form code words of fixed length. The above procedure describes the approach of conventional error-detecting and error-correcting codes[39].

However, the success of these codes depends on the prior determination of the correct timing or synchronization. As Golomb[25] points out, there is a hierarchy of synchronization problems. Roughly, one may consider three ranges :

(i) symbol sync,

(ii) word sync and

(iii) frame sync.

In this study it will be assumed that symbol sync, or bit sync, has been achieved. In addition, a frame will be one word length and hence, in this case, word sync and frame sync coincide. Even after symbol sync is attained, the decoder must frame the received symbols into correct blocks or words. If it incorrectly frames the received symbols, serious errors may result, even in the absence of additive errors.

The receiver may fail to obtain word sync if it does not know when transmission commenced. In another situation, the transmitter and receiver may be initially synchronized and then loose sync due to either the insertion or deletion of symbols from the correct se-

quence[48,59,60].

In this study, the effect of additive error and loss of word sync on cyclic codes will be examined. It will be assumed that no insertions or deletions of symbols occur in the transmitted sequence, and that sync loss means the misframing of the sequence by the transmitter. However, it should be acknowledged that after the passage of more than a word length, a single occurrence of either type of sync error is indistinguishable.

For a convenient notation, the word "slip" will be defined to mean loss of synchronization, in the sense used in this study. Also, the word "sync" is a widely accepted abbreviation for synchronization.

The motivation for choosing cyclic codes is that their additive-error-detecting and correcting properties have been studied extensively[39], especially the important Bose-Chaudhuri-Hocquenghem or BCH codes[6,7,30]. In addition, as mentioned earlier, cyclic codes are relatively easy to encode and decode[39]. As they stand, however, cyclic codes are very susceptible to sync errors, or slip[38,52]. For example, if any cyclic code word from a cyclic code of word length n is transmitted twice consecutively, then any n-sequence from this 2n-sequence will be a code word from this cyclic code. The object of this study is to modify the cyclic codes so that they can detect and correct slip without destroying their other desirable properties.

Two possible physical causes of sync error, or slip are

(i) receiver uncertainty as to the exact time that transmission commences and

(ii) the receiver losing count of the received bits, due to a malfunction.

A more detailed examination of the possible causes of sync error, and in general a good

introduction to the problem of loss of sync, is given in the panel discussion on "Synchronization" by Golomb et al[25].

1.3 A Brief Review of Methods of Synchronization

A conceptually straightforward method of maintaining synchronism is to introduce a special synchronizing symbol which would be used for this purpose only. Examples[22, 25] are the letter space in Morse code and the start and stop pulses in teletype. A major disadvantage of this method is that it makes inefficient use of the channel, since one (or more) of the symbols is used for synchronization only. A practical consideration is that the system has to generate and recognize an extra symbol. In addition, false sync might occur if additive noise changes one of the other code symbols into the sync symbol.

Another method of maintaining sync is to attach a synchronizing sequence[3, 22,44] which uses only ordinary code symbols at the beginning or end of each code word. For instance, in a binary system, the sync sequence would be a known sequence of binary digits. Some sequences are better for this purpose than others[3,22] and a desirable feature is that the correlation of the sequence with a shifted version of itself should be small. Sequences which have this property are the Barker Sequences[3] and the Pseudo-Random Sequences[24, 26]. In general, a longer sequence gives greater protection against slip but it increases the redundancy of the code.

In some codes which use a sync sequence, the remaining bits in the code word are unrestrained[3]. In this case, unless the length of the sync sequence is greater than half a word length, the sync sequence itself may occur elsewhere in the code word. However, the code is able to detect slip of magnitude less than the length of the sync sequence. On

the other hand, some codes restrain the remaining bits so that the sync sequence can not occur elsewhere (assuming no additive noise). These codes have been called prefixed comma-free codes[44], or prefix synchronized encodings[22] since they form a subset of the general class of comma-free codes[24]. In a comma-free code, loss of sync always results in a sequence that is not a valid code word. However, if additive noise is present, it is desirable that the sync sequence differ from neighbouring sequences in the code word by more than one bit[3, 44]. This gives some protection against the noise and still allows the number of code words to be large.

At present, methods are known for finding or estimating the number of code words in comma-free codes[10, 11, 24, 32] or prefixed comma-free codes[22, 44], but little is known about their additive-error-detecting or correcting properties, nor are general synthesis procedures available for constructing these codes. Moreover, during periods when no slip occurs, the bits in the sync sequence cannot be used for additive detection or correction. In passing, it should be mentioned that the sync bits may be distributed throughout the code word[47] (interlaced) rather than grouped into a sequence.

The above discussion has implicitly assumed that no bits are lost or gained in the transmitted sequence, i.e., the number of bits which arrive at the receiver is equal to the number of bits transmitted. The loss or gain of one or more bits in the sequence can result in a loss of sync. For example, assume that an n-bit word loses b bits, and that the first bit of the shortened word arrives in synchronism. The receiver will then count n bits and try to decode the received n-tuple. However, this n-tuple will contain b bits from the following n-tuple. The next n-tuple framed will then be b bits out of sync. If no corrective action is taken, from here on the receiver and transmitter would be b bits out of sync, (if no more bits are lost or gained). Hence, after the initial word with b bits

missing, the problem is the same as if the receiver was turned on and started to decode b bits out of sync (in the same direction). If instead of a loss of b bits, there was a gain of d bits, the receiver and transmitter would be out of sync by d bits in the other direction.

In studying the problem of slip for group codes, it is necessary to specify if one or both of these sources of sync error are present. Sellers[48] and Ullman[59, 60] have studied the case where bits are lost or gained in the sequence.

Sellers[48] describes a block code that will correct an error consisting of a gain or loss of a bit within the block. His code can be generalized to correct the loss or gain of a burst of bits. In addition, the code can correct additive errors in the vicinity of the bit loss or gain[48]. The code is constructed by inserting a synchronizing sequence* into a burst-error-correcting code at periodic intervals. The synchronizing sequence locates the approximate position of the bit loss or gain. At the location a bit is inserted or removed from the block, depending on whether a loss or a gain has occurred. The code then corrects the erroneous bits between where the error occurred and where the correction took place.

Ullman[59,60] has also studied sync error which results from the loss or gain of a burst of bits. Ullman describes a block code which corrects a single sync error per block (i.e. insertion or deletion of a single bit per block). He shows that this code has, at most, three bits more redundancy than that of an optimal code for this class of errors. The codes have specified positions for information, and no table look-up is necessary to encode or decode. The codes are not group codes.

Ullman has also shown that the minimum redundancy necessary for a code of block length p to correct the loss or gain of a single bit approaches $1 + \log_2 p$, as p approaches infinity. His codes have a redundancy less than $4 + \log_2 p$ and hence have three or less bits

---

* Sellers calls this synchronizing sequence a "character".

more than the optimum. By comparison, the codes of Sellers have redundancy at least $(12p)^{1/2} - 3$. Levinstein (1965)[37] has also described a code similar to that of Ullman's.

In this dissertation it is assumed that there are no bit losses or gains (deletions or insertions) and that "sync error" or "slip" means that the receiver frames the wrong n-tuple in the sequence from a block code of length n.

The concept of a comma-free code was mentioned earlier in this section. It will be recalled that a code, of word length n, is said to be comma-free if for all non-zero values of slip, the framed n-tuple is not a code word (assuming a noiseless channel).

Golomb, Gordon and Welch[24] determined the greatest number of words that a comma-free code (or dictionary) can possess. They proved that for a comma-free code having word length n and q symbols (or states), the maximum number of words, $W_n(q)$ is upper bounded by

$$W_n(q) \leq \sum_{b|n} \mu(b)\, q^{n/b}$$

where the sum is over all divisors b of n, and $\mu(b)$ is the Mobius function :

$$\mu(b) = \begin{cases} 1 & \text{if } b = 1, \\ 0 & \text{if b has any square factor} \\ (-1)^r & \text{if } b = p_1 p_2 \cdots p_r, \text{ where } p_1, \ldots p_r \\ & \text{are distinct primes.} \end{cases}$$

Eastman[11] has shown that this upper bound can be achieved for all odd values of n. Jiggs[32] has listed the known (1963) comma-free dictionary sizes for even n. For binary channels (q = 2), a simpler, but looser bound is given by[22]

$$W_n(2) \leq 2^n/n,$$

which implies that[52] $k \leq n - \log_2 n$, where k is the number of information bits. In a later

paper, Golomb[32] introduces the concept of "comma-free codes of index r", which applies to a code where for every slip, the framed n-tuple differs from a code word in at least r positions. Ordinary comma-free codes are of index 1. The usefulness of the extension to codes of index r for noisy channels is clear. Further work on comma-free codes and various extensions are also listed in the bibliography[10, 11, 12, 13, 23, 27, 45].

It appears that at present not much is known about the error detecting and correcting properties of comma-free codes in general. Observe that a comma free code cannot be a group code, since the zero word is excluded from comma-free codes. However, Stiffler[52] has shown that, given any (n,k) binary cyclic code*, there will exist an (n,k) coset code which is comma-free if and only if $k \leq (n-1)/2$ (see also Corollary 2.1 in this thesis). Stiffler appears to be the first to recognize the advantages of using coset codes for situations where both additve errors and slip occur. Some of the advantages of coset codes are (1) no additional redundancy is necessary, (2) they are simple to encode and decode and (3) the minimum distance between code words is not reduced. Stiffler showed that a group code has at least one coset which can detect s bits of slip if three vectors in its null space satisfy a set of conditions[52]. This result seems to be awkward to apply to arbitrary group codes, but fortunately it simplifies considerably for cyclic codes. The results obtained by Stiffler on cyclic codes are examined in detail in Chapter II and will not be repeated here.

---

* An (n,k) cyclic code has k information bits and n-k parity check bits. This is also true for an (n,k) coset code.

However, his results are of limited value for a noisy channel (additive noise) since the decoder cannot distinguish between slip and additive error. Recall that for a comma-free code, the misframed n-tuple is only guaranteed to be distance one from a valid code word. Hence, some combination of slip plus a single additive error could generate a valid code word. An interesting feature of Stiffler's work[52] is that he uses vector-matrix methods rather than polynomial algebra. See Chapter IV of this thesis for a development and extension of the vector-matrix approach for coset codes of cyclic codes. (See also the publication by the author[55]).

A later paper by Frey[18] also uses coset codes of cyclic codes to detect slip and additive error, but he does not seem to be aware of Stiffler's earlier work. Frey's results are also restricted to detection. He uses the polynomial representation of code words. A certain amount of slip correction is achieved by having the decoder detect slip error and adjust the framing until a valid n-tuple is framed. If both slip and additive error can occur, the search may be long. To avoid a long search, Frey introduces a short synchronizing sequence.

A broader concept than comma-free codes, and in fact broader than comma-free codes of index r, was introduced by Levy[38]. He defined a pair of numbers $[S,\delta]$ for a block of code length n, where a slip of S bits or less would produce an n-tuple which differed from a legitimate code word in at least δ places. If $S = [n/2]$*, then a block code with slip detecting characteristic $[S, \delta]$ becomes a comma-free code with index of comma freedom δ. By intuition and trial and error, Levy discovered a set of coset codes (which he called altered cyclic codes) of cyclic codes which displayed the $[S, \delta]$ characteristic. He admits that the coset codes found by him may not be optimum (Tong[58] improved on Levy's results) and he gave no general method for extending the list for larger δ. Observe that for

* $[x]$ means the integral part of x.

a block code with characteristic $[S, \delta]$, if there is a slip of S bits or less, the framed n-tuple will not be a code word even in the presence of $\delta - 1$ or less additive errors. Hence, such a code can detect the simultaneous occurrence of S or less bits of slip and $\delta - 1$ or less additive errors. However, the decoder will not be able to determine if the error is due to slip only, or additive error only, or both simultaneously. It is seen that Levy's results are essentially for error detection, although by searching, the decoder may be able to recover sync when relatively few additive errors are present. When correct sync is attained, the number of errors will be observed to fall sharply, with high probability[52].

Tong[58] extends the previous work on the synchronization of coset codes of cyclic codes and shows that they can not only detect slip but also correct it. He shows that the ability of an (n,k) coset codes to correct slip cannot exceed $(n-k-1)/2$.* He also presents a class of coset codes which can distinguish between slip errors and additive errors and which can correct both types if they do not occur simultaneously in the same n-tuple. In this thesis, a class of coset codes is presented which can correct the simultaneous occurrence of slip errors and additive errors. For cyclic codes with some special properties, Tong constructs coset codes of the given code so that one bit of slip can always be corrected.

Tong also considers shortened cyclic codes. He shows that, given a binary cyclic code which corrects $e \geq 2p+1$ errors, $p > 0$, this code can be shortened by $2p+1$ bits or more to give a code which can correct p bits of slip and also has expected noise tolerance :

$$E(\beta) = 2^{-2\beta} \sum_{i=1}^{2\beta} (e - i) \binom{2\beta}{i-1}, \qquad 0 < \beta \leq p ,$$

---

* However, in Theorem 2.5 in Chapter II, a condition is imposed on this result.

where a code is said to have expected noise tolerance of $E(\beta)$ bits; if with probability of at least 1/2, the code can correct p bits of slip with $E(\beta)$ or fewer additional additive errors in the received word. He develops a similar technique for codes which can correct more than one error. It is shown that by shortening the code by $2p+1$ bits or more and then adding p zeros to each end of the shortened word that the code can be made to correct p or less bits of slip. Tong's work will be discussed further in the text.

Caldwell[8] and Bose and Caldwell[5] also examine the problem of slip in a noisy channel. Their method is not based on coset codes but on a new technique. In words, the technique consists of adding specified bits to either end of a cyclic code word and restraining some of the information bits in the code word. These "extended" codes have the ability to correct the simultaneous occurrence of slip and additive error without sacrificing any additive error correcting ability. However, the redundancy of the code is increased. Caldwell's technique is examined further in Chapter VII where some new results are also presented. Weldon[61] has shown that Caldwell's technique can be used with any additive-error-correcting cyclic code.

The following authors among others, have also examined the problem of synchronization of codes : Hackett[28], Kasahara and Kasahara[33,34], Schutzenberger[46] and Stiffler[54].

1.4 Notation and Definitions

Cyclic codes are adequately described in the literature[39, 40, 41] and their description here will be brief. An (n,k) cyclic code will mean a cyclic code with word

length n having k information bits and n-k parity check bits. A cyclic code is a linear code with the added property that any cyclic permutation of the digits in a code word is also a cyclic code word. To illustrate, if the n-tuple $(a_1, a_2, \ldots a_n)$ represents a cyclic code word, then any cyclic shift of the digits, such as $(a_{i+1}, a_{i+2}, \ldots, a_n, a_1, a_2, \ldots a_i)$ is also a cyclic code word for all i. If the a_i's belong to a finite field F of q elements, an (n,k) cyclic code has q^k code words*.

The sum of two cyclic code words $(a_1, a_2, \ldots, a_n)$ and $(b_1, b_2, \ldots b_n)$ is also a cyclic code word (group property) and is given by $(d_1, d_2, \ldots, d_n)$ where $d_i = a_i + b_i$, i = 1 to n, and the rules of the addition are determined by the field F. For example, for binary codes, addition and multiplication are performed in modulo 2 arithmetic, where 0 + 0 = 1 + 1 = 0 and 0 + 1 = 1 + 0 = 1 for addition, and 0 x 1 = 1 x 0 = 0 and 1 x 1 = 1 for multiplication.

The n-tuple $(a_1, a_2, \ldots, a_n)$ may be regarded as[2, 39]

(i) an n-vector and

(ii) as a polynomial over F, whose coefficients are the a_i's .

If the code words are viewed as n-vectors, the cyclic code becomes a subspace of the linear vector space of all n-vectors whose elements belong to the finite field F. In addition, the vectors will possesss the cyclic property described above. If, instead, the cyclic code words are represented by polynomials, or more precisely, as elements of the algebra of polynomials modulo $x^n - 1$, corresponding to each n-tuple $(a_o, a_1, a_2, \ldots, a_{n-1})$ there is a polynomial** $a_o + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1}$, where the $a_i \in F$. It can be shown that the subspace is

---

* This statement is also true of any (n,k) linear code whose elements come from a finite field of q elements.

** The n-tuple begins with a_o instead of a_1 to fit accepted polynomial notation.

a cyclic subspace (or cyclic code) if and only if it is an ideal in the algebra of polynomials modulo $x^n - 1$ [2, 39]. The generator of the ideal, G(x), is called the generator polynomial of the cylic code. For an (n,k) cyclic code, G(x) has degree (n-k) and divides $x^n - 1$. Any cyclic code word may be written as Y(x) G(x), where the polynomial Y(x) has degree less than k. The quotient $H(x) = (x^n - 1)/G(x)$ is called the recursion polynomial of the code.

Given any polynomial P(x), let its syndrome[58], written $\{P(x)\}$, be defined as the remainder after division by G(x). From the previous definition of a cyclic code, it follows that the syndrome of a polynomial is zero if and only if it is a cyclic code word.

An (n,k) coset code[39] can be obtained from an (n,k) cyclic code (group codes in general) by adding a fixed n-tuple to each cyclic code word. If this n-tuple is itself a cyclic code word, then the original cyclic code is regenerated. There are q^{n-k} distinct coset codes of an (n,k) cyclic code with q symbols, and each coset code has q^k code words. One of the cosets is the original cyclic code, obtained by adding the zero n-tuple. Any coset code word can be written (algebraically) as W(x) + C(x) where W(x) is a cyclic code word and C(x) is the added polynomial. For any non trivial coset code, C(x) is not divisible by the generator polynomial of the cyclic code, G(x). In fact, since the syndrome of a cyclic code word is zero, it follows that

$$\{W(x) + C(x)\} = \{C(x)\} .$$

The vector-matrix description and the polynomial description are essentially equivalent although the language may be quite different. Many engineers are not familiar

with modern algebra but they usually have some knowledge of matrix theory. However, since much of the work on coding uses the polynomial representation, the matrix description is also presented in this thesis (see Chapter IV). Both descriptions are useful and there is some advantage in having two points of view. Hence, loss of synchronization, or slip, will be examined using both representations.

In this study, some of the results apply only to binary codes and some generalize to any finite field F. Whenever a result is valid only for binary codes, this fact will be stated explicitly.

Consider any three consecutive transmitted code words from a block code of length n

$$(a_1, a_2, \ldots, a_n) (b_1, b_2 \ldots , b_n) (d_1, d_2, \ldots, d_n) . \quad (1.1)$$

If the receiver is correctly synchronized with the transmitter, it will frame the n-tuples correctly as shown in (1.1). However, if slip has occurred, and the n-tuple under consideration is $(b_1, b_2, \ldots b_n)$, the decoder may frame either

$$(b_{s+1}, b_{s+2}, \ldots, b_n, d_1, d_2, \ldots d_s) , \; s \leqslant n/2 \quad (1.2a)$$

or

$$(a_{n-s+1}, a_{n-s+2}, \ldots a_n, b_1, b_2, \ldots b_{n-s}) , \; s < n/2 \quad (1.2b)$$

Case (1.2a) will be called a slip to the right of s bits (right slip) and case (1.2b) a slip to the left of s bits (left slip). In Figure 1.1, it is assumed that s bits of left slip have occurred, where $\overline{A}$ is the vector representation of $(a_1, a_2, \ldots a_n)$, etc. If there were no slip, the receiver would frame the word $\overline{B}$. As explained in more detail in Section 1.6, if $\overline{A}$, $\overline{B}$ and $\overline{D}$

are cyclic code words, $\overline{C}$ is the fixed n-tuple added to each code word before transmission. In the figure $\overline{A}_2$ represents the s bits from $\overline{A}$ which enters the receiver frame.

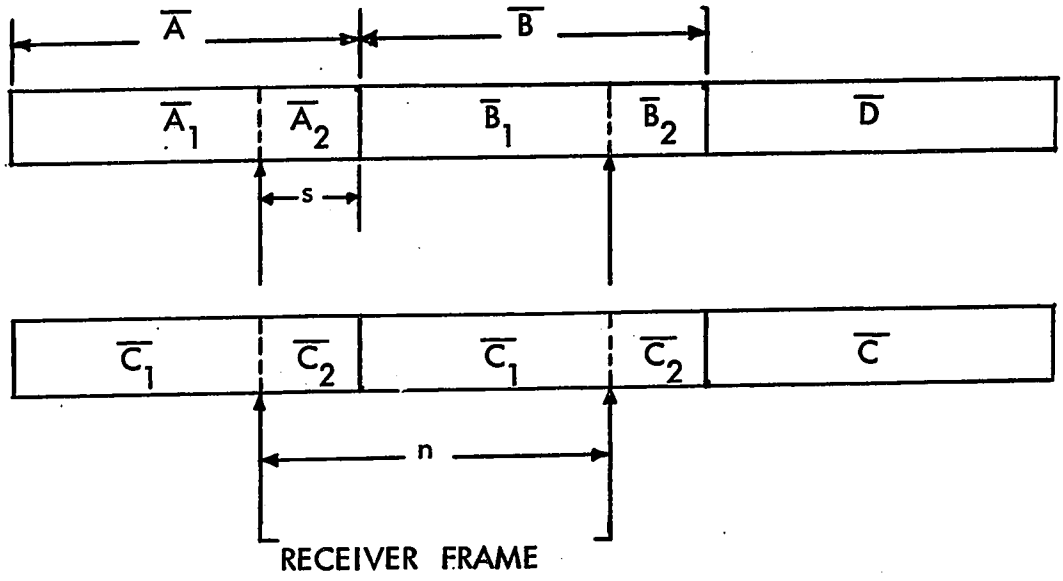

FIGURE 1.1 Receiver Frame for s Bits of Left Slip.

The amount of slip is usually considered to be less than half of a word length. If n is even and slip is n/2 bits, right slip is arbitrarily assumed. If the n-tuples in (1.2a) and (1.2b) are not code words for all right and left slip less than or equal to S bits ($S \leq n/2$), the code will be said to have comma-freedom* S. If the framed n-tuple is not a code word for all

---

* Tong[57] calls this "comma-free freedom S".

non-zero values of slip, the code is said to be comma-free[22]. Also, if (1.2a) and (1.2b) are not code words for right and left slip equal to s, for any three adjacent code words (not necessarily different), the code is said to be invulnerable to slip s. Note that invulnerability to slip s does not necessarily imply that the code is invulnerable to lesser values of slip.

An important concept in the study of block codes is the distance (Hamming distance[29]) between two code words. This may be defined as the number of positions in which corresponding elements are different. A related concept is the weight of a code word which is defined as the number of non-zero elements in the code word. For example, the distance between the 5-tuples (10011) and (01101) is 4 and both have weight 3. For any group code, it is easily shown that the minimum distance, d, is equal to the minimum weight of the code. This property does not necessarily hold for non-group codes. There is also another metric called the Lee distance[36], but it will not be considered in this study.

1.5 Mathematical Description of Slip

Using the polynomial representation, associate with each n-tuple $(a_o, a_1, \ldots, a_n)$ the polynomial $a_o + a_1 x + \ldots + a_{n-1} x^{n-1}$. Let A(x) represent a code word from an (n,k) cyclic code, and write

$$A(x) = a_o + a_1 x + \ldots + a_{n-1} x^{n-1} \tag{1.3}$$

where the coefficients a_i are elements of the finite field F. Consider the product *

---

* Given a polynomial of degree greater than or equal to n, the desired polynomial is obtained by division by $x^n - 1$, and selecting the remainder. This is equivalent to applying the relation $x^n = 1$ to the given polynomial.

$$x\,A(x) = x\,(a_o + a_1 x + \ldots a_{n-1} x^{n-1})$$

$$= a_{n-1} + a_o x + a_1 x^2 + \ldots + a_{n-2} x^{n-1}. \qquad (1.4)$$

Hence, it is seen that multiplication by x is equivalent to a cyclic shift of one bit. Similarly, multiplication by x^{-1} is equivalent to a cyclic shift of one bit in the opposite sense. Now, consider the n-tuple in (1.2b) and assume s bits of left slip. The polynomial representation of this n-tuple may be written as

$$x^s B(x) + U_s(x) \qquad (1.5)$$

where B(x) represents the n-tuple $(b_1, b_2, \ldots b_n)$, and $U_s(x)$ is a polynomial of degree less than s whose coefficients are unknown elements of the field F. At this point it is worthwhile to mention an awkward situation due to the polynomial representation. From the n-tuple in (1.2b), it is seen that the s elements $(b_{n-s+1}, \ldots, b_n)$ are lost at the decoder. However the operation $x^s B(x)$ cyclically shifts them around to occuply the first s positions in the framed n-tuple. Hence the polynomial $U_s(x)$ represents the sum of the s-tuples $(b_{n-s+1}, \ldots., b_n)$ and $(a_{n-s+1}, \ldots.., a_n)$. Since the latter s-tuple comes from an unknown left-adjacent word, the coefficients of $U_s(x)$ may be regarded as random.

Recalling the n-tuple in (1.2a) the polynomial representation of s bits of right slip may be written as

$$x^{-s} B(x) + U_{-s}(x) \qquad (1.6)$$

where $U_{-s}(x)$ is a random polynomial whose lowest term has degree greater than or equal to n-s. It can be verified that (1.6) may be written as

$$x^{n-s}\ B(x) \ + \ x^{n-s}\ U_s(x) \qquad (1.7)$$

or

$$x^{n-s}\ (\ B(x) \ + \ U_s(x)\ )\ . \qquad (1.8)$$

To keep the discussion at this stage simple, a noiseless channel has been implicitly assumed, i.e., no additive errors can occur. In later chapters, noisy channels will be considered and the effects of additive errors will be examined. The description of slip in this section is algebraic and the vector-matrix description of slip will be given in Chapter IV.

1.6 Description of The Transmitted Code Word

As was mentioned earlier, cyclic codes as such are vulnerable to synchronization error, hence it is desirable to alter the cyclic code words before transmission. One procedure is to add a constant polynomial C(x) to each cyclic code word, W(x), before transmission. The transmitted word is then

$$\begin{aligned} B(x) &= W(x) + C(x) \\ &= Y(x)\ G(x) \ \dot{+}\ C(x) \end{aligned} \qquad (1.9)$$

The new code is called a coset code[39, 57] and has the same number of words as the original cyclic code. Depending on the choice of C(x), the coset code may or may not have good slip detecting or correcting properties.

Another procedure[5, 8, 61] is to restrain m of the k information bits, $m < k$, of the cyclic code words to give the new code words

$$D(x) \ = \ (\ J(x)\ F(x) \ + \ 1\ )\ G(x) \qquad (1.10)$$

where F(x) is a primitive polynomial of degree m and J(x) is an arbitrary polynomial of degree less than k - m. The new code is no longer cyclic (in fact it is not even a group code) and has q^{k-m} words. It is a subset of the original cyclic code since each word, D(x), is actually a cyclic code word. These codes are also able to detect and correct slip. There is another class of codes which is obtained by combining the coset code technique and the above technique. These codes will be studied in Chapter VII.

None of the above techniques alter the length of the code word, n, but there are others[5, 58 61] which do. In this study, attention will be restricted almost exclusively to techniques that do not alter the length of the cyclic code. However some of the other techniques, briefly reviewed in Section I.3, will be examined again in Chapter VII.

I.7 Consideration of Additive Noise

Interesting results have been obtained by assuming that slip is the only source of error[23, 24, 32]. Such a channel is noiseless, in the additive error sense. These results, however, are of limited value for practical channels. For noisy channels it is desirable to know the combined effect of slip and additive error. Two widely used models for additive error are

(i) the noise affects the digits of the code words independently and,

(ii) the errors are dependent and tend to occur in bursts or clusters.

For some channels, such as telephone lines[1, 17], burst errors may give a better model for the experimental data, although a combination of the two types would given an even more accurate description. In later chapters, both independent and burst errors will be examined

in the discussion of slip for a noisy channel. Although most of the results presented in this study are valid for the general class of cyclic codes, in Chapter VI, the effect of slip in a channel with burst errors is studied for the Fire codes[16], which are a subset of the class of cyclic codes.

It might be added here that one of the main results of this study[55, 56, 57] is the demonstration of the existence of a class of coset codes of cyclic codes which can correct both slip and additive error, even when they occur simultaneously in a received word.

CHAPTER II

LOSS OF SYNCHRONIZATION IN A NOISELESS CHANNEL FOR COSET CODES

2.1 Introduction

In this chapter, slip is assumed to be the only source of error. Although the results obtained do not apply to noisy channels, they are useful for theoretical purposes and give considerable insight into the more general situation where additive errors are also present. In addition, the methods of the analysis can be extended to the noisy channel. All the results of this chapter are valid for cyclic codes whose elements belong to any finite field F. Both detection and correction of slip are examined and the meaning of "correction" is discussed in some detail. Some of the results presented in this chapter are new and will be indicated as such where applicable.

As stated in Chapter I, the transmitted words are coset code words which are obtained by adding a fixed polynomial C (x) to each cyclic code word W (x), before transmission. From each received word V (x), the decoder subtracts the polynomial C (x). If there is no slip, the received word minus C (x) , i.e., V (x) - C (x), is the original cyclic code word. If slip did occur, V (x) - C (x) may not be a cyclic code word, depending on C (x) and the amount of slip. Since the decoder can detect slip only when the received word minus C (x) is not a cyclic code word, C (x) should be chosen so that V (x) - C (x) is not a cyclic code word for the maximum amount of slip. To detect slip, the decoder divides V (x) - C (x) by the generator polynomial G (x) of the cyclic code. Since a polynomial is a cyclic code word if and only if it is

divisible by G (x), the decoder decides that slip is present if a non-zero remainder is obtained. If the remainder is zero, either there is no slip present or the decoder has failed to detect it. Equivalently, recalling that the remainder after division by G (x) is called the syndrome, the decoder can detect slip if and only if the syndrome is not zero. Note that this discussion assumes that the decoder is analyzing a single word length. Results on the detection of slip are presented in the next section.

2.2 Detection Of Slip

This section presents some results on the detection of slip in a noiseless channel. The transmitted words are from a coset code which is derived from a cyclic code, and, as explained in the previous section, the decoder detects slip by computing the syndrome. The result which follows was first proved by Stiffler[52], and is also stated by Levy[38] and Tong[58].

Theorem 2.1

Given any (n, k) cyclic code, there exists an (n, k) coset code which can detect all slip less than or equal to $n - k - 1$ bits.

Proof : First, note that the coset code satisfying the theorem is not unique for a given cyclic code. To prove the theorem, we make a simple choice and at the end of the proof some other possible choices are mentioned. In order to construct the coset code, add the polynomial $C(x) = 1$ to each cyclic code word $W(x)$ before transmission. Assume

that s bits of left slip occur at the receiver. The decoder subtracts C (x) from the received n - tuple to obtain

$$x^s (W(x) + C(x)) + U_s(x) - C(x) \tag{2.1}$$

where $U_s(x)$ is a polynomial of degree $s - 1$ or less whose coefficients are random. $U_s(x)$ accounts for the unknown s bits which come from the left - adjacent word to W (x). In order to determine if the polynomial in (2.1) is a cyclic code word, the decoder divides by the generator polynomial G (x) and examines the remainder, or syndrome. Denoting the syndrome of a polynomial P (x) by $\{P(x)\}$, the decoder can detect slip if and only if

$$\{x^s (W(x) + C(x)) + U_s(x) - C(x)\} \neq 0 \tag{2.2}$$

since if (2.1) is a cyclic code word its syndrome is zero and the decoder concludes that no slip has occurred. Since W (x) is a cyclic code word, $\{x^s W(x)\} = 0$ for any s and 2.2 becomes *

$$\{C(x)(x^s - 1) + U_s(x)\} \neq 0. \tag{2.3}$$

Written as an n - tuple, the polynomial in (2.3) becomes

---

* In general, $\{P(x) + Q(x)\} = \{P(x)\} + \{Q(x)\}$ and $\{P(x)\,Q(x)\} = \{\{P(x)\}\{Q(x)\}\}$ where P (x) and Q (x) are arbitrary polynomials.

$$(u'_o , u_1 , u_2 , \ldots u_{s-1} , 1, 0, \ldots , 0) \tag{2.4}$$

where $u'_o = u_o - 1$ since $C(x) = 1$ and

$$U_s(x) = u_o + u_1 x + \ldots + u_{s-1} x^{s-1} . \tag{2.5}$$

It is clear that the n - tuple in (2.4) is a burst of length $s + 1$ or less*. But, for an (n , k) cyclic code, it is known that a burst of length $n - k$ or less cannot be a cyclic code word unless it is the zero word.[39] However, (2.4) cannot be the zero word since $U_s(x)$ cannot cancel the term x^s, although $-C(x) = -1$ itself may be cancelled by $u_o = 1$. It can now be concluded that (2.4) is not a cyclic code word if $s + 1 \leqslant n - k$, i.e., if $s \leqslant n - k - 1$.

If right slip of s bits is assumed, (2.3) would have the form

$$\left\{C(x)(x^{-s} - 1) + U_{-s}(x)\right\} \neq 0 \tag{2.6}$$

for all $U_{-s}(x)$, where

$$U_{-s}(x) = u_{n-s} x^{n-s} + u_{n-s+1} x^{n-s+1} + \ldots + u_{n-1} x^{n-1} \tag{2.7}$$

and writing the polynomial in (2.6) as an n - tuple we have

$$(-1 , 0, \ldots , 0 , u'_{n-s} , u_{n-s+1} , \ldots , u_{n-1}) \tag{2.8}$$

where $u'_{n-s} = u_{n-s} + 1$. By performing a cyclic shift of one bit on (2.8) it is seen that

---

* A burst of length b may be defined as an n - tuple which is zero everywhere except in an interval of length b .

(2.8) is a burst of length $s+1$ or less, and hence can not be a cyclic code word if $s+1 \leq n-k$, i.e., if $s \leq n-k-1$. It follows that the decoder can detect all slip less than or equal to $n-k-1$ bits.

Q.E.D.

In the above proof, the coset was obtained by choosing $C(x) = 1$. In fact, the choice $C(x) = \alpha$ or $C(x) = \beta x^{n-1}$ where α and β are arbitrary elements of the coefficient field F, would give the same result. The proof given here differs somewhat from existing proofs in that explicit use has been made of the fact that a burst of length $n-k$ or less cannot be an (n, k) cyclic code word.

The proof of Theorem 2.1 may seem rather long, but this is partly because new terms and operations were introduced during the proof. In later proofs, it will be assumed that the reader is familiar with the proof of Theorem 2.1. To gain further insight, Theorem 2.1 will now be proved in a slightly different manner[58]. Starting at (2.3) and substituting $C(x) = 1$, we have

$$\left\{x^s + U_s(x) - 1\right\} \neq 0. \tag{2.9}$$

Since $U_s(x)$ has a degree less than s, it can not eliminate the term x^s in (2.9) if $s \leq n-k-1$, because in this case, $\left\{x^s\right\} = x^s$. Therefore, when $s \leq n-k-1$ the inequality in (2.9) holds and the decoder can detect left slip. A similar proof holds for right slip.

Another way of stating Theorem 2.1 is to say that there exists an (n, k) coset code which has comma-freedom $n-k-1$. The next result shows that the per-

formance of Theorem 2.1 cannot be exceeded.[52,58]

Theorem 2.2

Given any (n , k) cyclic code, none of its coset codes has comma-freedom exceeding n - k - 1.

Proof : Allowing C (x) to be arbitrary, the decoder can detect s bits of left slip if and only if (see 2.3))

$$\left\{C(x)\,(x^s - 1) + U_s(x)\right\} \neq 0 \qquad (2.10)$$

for all $U_s(x)$. The above relation can be written as

$$\left\{C(x)\,(x^s - 1)\right\} \neq \left\{U_s(x)\right\} . \qquad (2.11)$$

Since the degree of the generator polynomial G (x) is n - k, the syndrome (i.e., the remainder) of any polynomial can have, at most, degree n - k - 1. Hence, regardless of the choice of C (x), if $s > n - k - 1$, there exists an $U_s(x)$ (since $U_s(x)$ is random) which is equal to $\left\{C(x)\,(x^s - 1)\right\}$. In other words, the inequality in (2.12) cannot be guaranteed for all $U_s(x)$ if $s > n - k - 1$. The proof is similar for right slip.

Q.E.D.

A corollary to Theorems 2.1 and 2.2 can be stated.[52,38,58]

Corollary 2.1

Given any (n , k) cyclic code, there exists a coset code which is comma-free if and only if $k \leq (n - 1)/2$.

Proof : Theorem 2.1 states that all slip less than or equal $n - k - 1$ can be detected. Observe that $k \leq (n - 1)/2$ implies that $k \leq n - k - 1$. A slip of $n - k$ or more in one direction can be considered as a slip of $n - (n - k) = k$ or less in the other direction. Since $k \leq n - k - 1$ here, the decoder can detect all slips. This proves that the coset code is comma-free if $k \leq (n - 1)/2$.

To prove the converse, assume that $k > (n - 1)/2$ which is equivalent to $k > n - k - 1$. This implies that there exists slip of s bits such that $k \geq s > n - k - 1$. But for such s, both s and $n - s$ are greater than $n - k - 1$. Hence, by Theorem 2.2 the corollary is proved.

Q.E.D.

Theorems 2.1 and 2.2 and Corollary 2.1 summarize the more important results relating to detection of slip by coset codes of cyclic codes in a noiseless channel. The ability of coset codes to correct slip in a noiseless channel will be investigated next.

2.3 Correction Of Slip

One definition of correction of slip is that the decoder can distinguish between right slip and left slip. In order for the decoder to do this, the syndromes for right slips must be different from the syndromes for left slips. However, this fact alone will not inform the decoder as to the number of bits of slip that has occurred. To compute the amount of slip, in addition to the direction, the syndromes for s_1 bits of right (left) slip must also be different from the syndromes for s_2 bits of right (left) slip if $s_1 \neq s_2$.

If the decoder can only distinguish between right slip and left slip, it can not recover sync on the next received code word without searching. A simple strategy for the decoder in this case is to reduce the slip by one bit for each new word that arrives until the correct sync is obtained. It is seen that if the slip is large, several words will be lost before the decoder resynchronizes. If, instead, the decoder corrects the slip by several bits at a time, it may cause slip in the opposite direction. However, on receiving the next code word it can adjust the sync in the other direction by a lesser number of bits. In this way it can gradually converge on the correct sync. On the other hand, if the decoder knows the amount and the direction of the slip, it can regain sync on the next word to arrive, without a search. Recovering sync without searching is certainly preferable to searching, but it results in a slightly reduced number of bits that can be corrected. In his work, Tong[58] generally uses correction of slip to mean the ability of the decoder to distinguish between right and left slip. On the other hand, Caldwell[8] and Bose and Caldwell[5] use correction to mean the ability of the decoder to compute both the magnitude and the direction of the slip. The distinction in the two meanings is worth pointing out since the first definition implies that the decoder must search to recover sync, whereas the second implies no searching.

The above discussion implicitly assumes that the decoder stores one word length at a time. However, there is another possibility, in that the decoder may store a longer sequence of bits. For example, it may store the preceding word while decoding the current word. In this case, if the decoder knows only the direction of the slip, it could search either "backward" in time, or "forward" as each succeeding bit arrives. This pro-

cedure will reduce the number of words lost before regaining sync. However, there are at least two penalties in this procedure. The first is that the decoder must store more data and hence will be more complicated and costly. The second is that it must search rapidly since data is arriving in a constant stream and large buffers may be necessary or overflow problems will arise.

It is even possible that the decoder may recover sync using slip detecting powers only, by performing an undirected search. Such a search may consist of merely allowing the framed n - tuple to slide along, bit at a time, as each succeeding bit arrives. The syndrome is computed for each n - tuple and sync is assumed when the syndrome is zero. In other words, if the current n - tuple is $(a_o , a_1 , a_2 , \ldots , a_{n-1})$ where a_o was the last bit to arrive, then if its syndrome is not zero, the decoder frames next $(a_{-1} , a_o , a_1 , \ldots a_{n-2})$, where a_{-1} arrives after a_o. In this way the decoder shifts bit by bit until the syndrome is zero. This method will always obtain the correct sync for a comma - free code but it may result in a false sync for codes which are not.

In the following theorem the decoder can only distinguish between right slip and left slip. Although the choice of $C(x)$ in this case is simple, it does not appear to have been stated elsewhere. It is readily derived from Tong's work.[58]

Theorem 2.3

Given any (n , k) cyclic code, there exists an (n , k) coset code which can distinguish between right slip and left slip if slip does not exceed $(n - k - 1) / 2$.

Proof : It is sufficient to show that the syndromes for right slip are different from the syndromes for left slip. As in Theorem 2.1, choose $C(x) = 1$ (or $C(x) = x^{n-1}$).

The requirement may be written as

$$\left\{x^s (W_1 (x) + C (x)) + U_s (x) - C (x)\right\} \neq \left\{x^{-r} (W_2 (x) + C (x)) + U_{-r} (x) - C (x)\right\} \quad (2.12)$$

for all $s, r \leq (n - k - 1) / 2$ and for all $U_s (x)$ and $U_{-r} (x)$ where $W_1 (x)$ and $W_2 (x)$ are any two cyclic code words. Putting all terms on the left and recalling that $\left\{x^s W_1 (x)\right\} = \left\{x^{-r} W_2 (x)\right\} = 0$, (2.12) reduces to

$$\left\{x^s C (x) + U_s (x) - x^{-r} C (x) - U_{-r} (x)\right\} \neq 0 \quad (2.13)$$

for all $s, r \leq (n - k - 1) / 2$. The polynomial in (2.13) is not a cyclic code word if and only if all of its cyclic shifts are not cyclic code words. Hence (2.13) is satisfied if and only if

$$\left\{x^{s+r} C (x) + x^r U_s (x) - C (x) - x^r U_{-r} (x)\right\} \neq 0 \quad (2.14)$$

where we have performed a cyclic shift of r bits by multiplying through by x^r. Recall from (1.7) that $U_{-r} (x) = x^{n-r} U_r (x)$ and hence $x^r U_{-r} (x) = U_r (x)$. Further, $x^r U_s (x) - U_r (x)$ may be written as $U_{s+r} (x)$. Applying these results to (2.14) gives

$$\left\{C (x) (x^{s+r} - 1) + U_{s+r} (x)\right\} \neq 0. \quad (2.15)$$

Comparing (2.15) with (2.3), it can be concluded that (2.15) is satisfied for all $U_{s+r} (x)$ if $s + r \leq n - k - 1$. Letting the maximum amount of slip in either direction be the same, the decoder can distinguish between left and right slip if slip does not exceed $(n - k - 1) / 2$.

Q.E.D.

It is instructive to point out that the decoder has more information at its disposal than was claimed by Theorem 2.3. In fact, it will be proved that if $C(x) = 1$, the decoder can distinguish between any two left slips but it can not do so for two distinct right slips. On the other hand, if $C(x) = x^{n-1}$, the decoder can distinguish between two right slips but not between two left slips. The case for $C(x) = 1$ will now be proved. Let left slips of s and r bits, where $s > r$, occur for any two codes words $W_1(x)$ and $W_2(x)$, respectively. It is required to show that

$$\left\{x^s (W_1(x) + C(x)) + U_s(x) - C(x)\right\} \neq \left\{x^r (W_2(x) + C(x)) + U_r(x) - C(x)\right\} \quad (2.16)$$

which reduces to

$$\left\{C(x)(x^s - x^r) + U_s(x) - U_r(x)\right\} \neq 0 . \quad (2.17)$$

Since $s > r$, let $U'_s(x) = U_s(x) - U_r(x)$, where $U'_s(x)$ is another random polynomial of degree less than s. Substituting $C(x) = 1$, (2.17) becomes

$$\left\{x^s - x^r + U'_s(x)\right\} \neq 0 . \quad (2.18)$$

Observe that $U'_s(x)$ cannot cancel the term x^s in the above expression if $s \leq n - k - 1$ (but it can cancel x^r), hence the decoder can distinguish between any two left slips not exceeding $n - k - 1$.

Now, let right slips of p and q bits occur, where $p > q$, for any two cyclic code words $W_1(x)$ and $W_2(x)$, respectively. To prove that the decoder cannot distinguish two right slips, by contradiction assume that

$$\left\{x^{-p} (W_1(x) + C(x)) + U_{-p}(x) - C(x)\right\} \neq \left\{x^{-q}(W_2(x) + C(x)) + U_{-q}(x) - C(x)\right\}, p>q \quad (2.19)$$

which can be reduced to

$$\left\{C(x)\,(x^{-p} - x^{-q}) + U_{-p}(x) - U_{-q}(x)\right\} \neq 0,\ p > q\ . \tag{2.20}$$

Performing a cyclic shift by multiplying through by x^{p}, (2.20) is equivalent to

$$\left\{C(x)\,(1 - x^{p-q}) + U_{p}(x) - x^{p-q}\,U_{q}(x)\right\} \neq 0\ ,\ p > q \tag{2.21}$$

where the relationship $U_{-p}(x) = x^{-p}\,U_{p}(x)$ was employed. The expression in (2.21) can be simplified further by setting $U'_{p}(x) = U_{p}(x) - x^{p-q}\,U_{q}(x)$ where $U'_{p}(x)$ has degree less than p. Substituting $C(x) = 1$, (2.21) becomes

$$\left\{1 - x^{p-q} + U'_{p}(x)\right\} \neq 0\ ,\ p > q\ . \tag{2.22}$$

It is clear that $U'_{p}(x)$ can cancel $1 - x^{p-q}$ for all $p \leqslant n - k - 1$, and hence the inequality (3.22) cannot be guaranteed for any such p. This proves that the decoder cannot distinguish between two right slips when $C(x) = 1$.

It can similarly be proved that when $C(x) = x^{n-1}$ the decoder can compute the magnitude of the slip for right slip but cannot do so for left slip. Finally, to distinguish between right slip and left slip, Theorem 2.3 requires that the slip $\leqslant (n - k - 1)/2$ bits. These results are summarized below.

Corollary 2.2

Given an (n, k) cyclic code, the coset code obtained by adding the polynomial $C(x) = 1$ can compute the magnitude and direction of all left slips $\leqslant (n - k - 1)/2$ bits, but can only determine the direction of the slip when right slip

$\leq (n - k - 1)/2$ bits occur . On the other hand, if $C(x) = x^{n-1}$, the decoder can determine the magnitude and direction of all right slips $\leq (n - k - 1)/2$ bits, but can only determine the direction of the slip when left slip $\leq (n - k - 1)/2$ occurs.

The implication of Corollary 2.2 is that for $C(x) = 1$, the decoder can recover sync on the next code word to arrive if left slip occurs, and for $C(x) = x^{n-1}$ it can recover sync on the next word if right slip occurs. It is believed that Corollary 2.2 has not been stated explicitly before. The next theorem, which is also believed to be new, covers the case where the decoder can recover sync on the next code word for both right and left slips.

Theorem 2.4

Given any (n , k) cyclic code, there exists an (n , k) coset code which can compute the magnitude and the direction (left or right) of the slip for all slips not exceeding $(n - k - 2)/2$.

Proof : The coset is obtained by letting $C(x) = x^{n-1} + 1$. Many of the steps in the proof can be borrowed from the proof of Theorem 2.3 and Corollary 2.2 since $C(x)$ was usually not specified until near the end.

First, it will be shown that the syndromes for right and left slips are distinct. This requirement is stated in (2.12), which reduces to (2.15). Substituting $C(x) = x^{n-1} + 1$ in (2.15) gives

$$\left\{(x^{n-1} + 1)(x^{s+r} - 1) + U_{s+r}(x)\right\} \neq 0 \qquad (2.23)$$

or

$$\left\{x^{s+r-1} - x^{n-1} + x^{s+r} - 1 + U_{s+r}(x)\right\} \neq 0. \tag{2.24}$$

Performing a cyclic shift of one bit by multiplying (2.24) by x gives

$$\left\{x^{s+r} - 1 + x^{s+r+1} - x + x\,U_{s+r}(x)\right\} \neq 0, \tag{2.25}$$

since $x^n = 1$. A convenient convention is now adopted for the random polynomial $U_p(x)$, where $p \leq n$. If another polynomial $Q(x)$ of degree q, where $q < p$ is added to $U_p(x)$, the sum is written as $U_p(x)$, i.e.,

$$U_p(x) + Q(x) \rightarrow U_p(x), \qquad p > q$$

since the sum is still a random polynomial of degree less than p. Adopting this convention, (2.25) can be written as

$$\left\{x^{s+r+1} - 1 + x\,U_{s+r}(x)\right\} \neq 0. \tag{2.26}$$

It follows that (2.26) is satisfied if $s + r + 1 \leq n - k - 1$ (recall proof of Theorem 2.3) or $s + r \leq n - k - 2$. Letting S be the maximum amount of slip in either direction, (2.26) is satisfied if $S \leq (n - k - 2)/2$.

The requirement that the syndromes for left slips s and r, where $s > r > o$ are to be distinct is given by (2.16) which reduces to (2.17). Substituting $C(x) = 1 + x^{n-1}$ in (2.17) gives

$$\left\{x^{s-1} + x^{s} - x^{r-1} - x^{r} + U_s(x)\right\} \neq 0, \qquad s > r > o$$

which simplifies to

$$\left\{ x^s + U_s(r) \right\} \neq 0 \ , \quad s > 2 \ . \tag{2.27}$$

It is seen that (2.27) is satisfied if $s \leq S \leq n - k - 1$. Similarly, for right slips p and q, where $p > q$, it suffices to show that (2.21) with $C(x) = 1 + x^{n-1}$ is satisfied, i.e., that

$$\left\{ x^{n-1} + U_p(x) \right\} \neq 0 \ , \quad p > q \tag{2.28}$$

and (2.28) is satisfied if $p \leq S \leq n - k - 1$. However, as shown earlier in the proof, it is required that $S \leq (n - k - 2)/2$.

Q.E.D.

The next question is whether the performance of Theorems 2.3 and 2.4 can be exceeded ? A partial answer to this question is given by the next theorem.[55]

Theorem 2.5

No coset of any (n , k) cyclic code, for which $2k \geq n$, can distinguish between right slip and left slip whenever slip exceeds $(n - k - 1)/2$.

Proof : The starting point will again be (2.12) in the proof of Theorem 2.3, and as before (2.12) reduces to (2.15) except that here C (x) can be arbitrarily chosen. Now, assuming that $s + r > n - k - 1$ and recalling that $U_{s+r}(x) = x^r U_s(x) - U_r(x)$, (2.15) can be written as

$$\left\{ C(x)\,(x^{s+r}-1) \right\} \neq \left\{ U_{s+r}(x) \right\} . \qquad (2.15')$$

The syndrome of a polynomial whose degree is less than $n-k$ is the polynomial itself, hence every such distinct polynomial has a distinct syndrome. Further, there are q^{n-k} distinct polynomials (q is the number of elements in the coefficient field F) of degree less than $n-k$ and hence such polynomials can generate all possible syndromes. If $s+r > n-k-1$, the degree of $U_{s+r}(x)$ can be as high as $n-k-1$, and if $U_{s+r}(x)$ is perfectly arbitrary (random) then it can, as explained, generate all syndromes. However, there are restraints on the arbitrariness of $U_{s+r}(x)$, since $U_s(x)$ and $U_r(x)$ are parts of coset words and so are not entirely arbitrary. In fact, for an (n, k) coset code (or cyclic code) there are k arbitrary information bits, and $n-k$ check bits which are determined once the information bits are chosen. It can now be seen that if $2k \geq n$, $U_{s+r}(x)$ will be completely arbitrary because $s \leq k$ and $r \leq k$. Hence, if $2k \geq n$ and $s+r > n-k-1$, $U_{s+r}(x)$ can generate all syndromes and so the inequality (2.15') cannot be guaranteed, regardless of the choice of $C(x)$.

Q.E.D.

Note also that $U_{s+r}(x)$ is completely arbitrary when $n-k-1 < s+r \leq 2k$ and both s and r do not exceed k. Hence, letting S represent the maximum slip in either direction, this condition becomes $(n-k-1)/2 < S \leq k$. The following can now be stated.[55]

Corollary 2.3

No coset of any (n, k) cyclic code, for which $3k \geq n$, can distinguish between right and left slip whenever slip S lies in the range $(n-k-1)/2 < S \leq k$.

Tong[58] states a theorem (his Theorem 3) which is the same as Theorem 2.5 except that the condition $2k \geq n$ is omitted. The absence of any condition implies that the adjacent words are arbitrary, but this is not the case for the noiseless channel which is assumed. In fact, the adjacent words are coset words.

This completes the discussion of slip for coset codes in a noiseless channel.

CHAPTER III

LOSS OF SYNCHRONIZATION FOR COSET CODES IN A NOISY CHANNEL

3.1 Introduction

In this chapter, it will be assumed that additive errors (due to additive noise) affect transmitted bits independently. As mentioned earlier, there is another important class of errors called burst errors, but these will be discussed in Chapters V and VI .

Additive errors will be represented by the polynomial $E(x) = e_o + e_1 x + \ldots + e_{n-1} x^{n-1}$. In a received n - tuple, if there is an additive error at position e_i , then $e_i \neq o$, but if not, $e_i = o$. In this way, the number of additive errors is equal to the number of terms in $E(x)$, which is called the weight of $E(x)$, written as $W[E(x)]$. For example, if there are e additive errors, then $W[E(x)] = e$.

Throughout the discussion, the minimum weight of the cyclic code under consideration will be designated by d , which is assumed to be a known quantity. The zero word is always a cyclic code word but is, of course, ignored when computing d . For cyclic codes (group codes in general), the minimum weight is equal to the minimum distance and d will represent both.

A code with minimum distance d can either detect $(d - 1)$ or less additive errors or correct $(d - 1)/2$ or less additive errors (ignoring slip for the moment).

If the decoder is designed to correct e or fewer additive errors, where $e < (d-1)/2$, then it can also detect all additive errors e' in the range $e < e' \leq d - (e+1)$. Given an (n , k) cyclic code, it can easily be shown that the minimum distance for all its coset codes is also d . Since a coset code is not a group code, its minimum distance and minimum weight need not be the same.

The discussion of slip in a noisy channel will be divided into two cases :

Case I : In this case, it is assumed that the two types of error do not occur simultaneously. In other words, if a given n - tuple is transmitted, it is assumed that it will arrive at the receiver with either additive errors or slip errors, but not both. Of course, some n - tuples may arrive with no errors at all. Physically, this implies that the probability of both types of error occurring in the same n - tuple is negligibly small. If the probability of occurrence of both types of error is small and they are independent, then the probability of their joint occurrence (i.e., the product of their individual probabilities) will be much smaller. Hence, although Case I is restricted, it can serve as a reasonable model for some low noise channels.

Case II : Here, additive error and slip are allowed to be present simultaneously in any n - tuple at the receiver. Case II is obviously more general than Case I and covers a wider class of channels. Case I is treated separately, however, because it is examined in the literature,[58] and also because it provides insight in solving the more general Case II.

Both Case I and Case II will further be broken down into two problems, namely, detection (A) and correction (B) .

3.2 Case I: Slip And Additive Error Do Not Occur Simultaneously

A. Detection

By Theorem 2.1, any (n, k) cyclic code has a coset code which can detect $n - k - 1$ or less bits of slip when no additive errors are present. However, since the coset code has the same minimum distance, d, as the original cyclic code, it can also detect $d - 1$ or less additive errors if no slip is present. Note that the decoder can detect that some kind of error is present, but cannot determine whether it is slip or additive error. The above discussion can be summarized in the following theorem.

Theorem 3.1

Any (n, k) cyclic code has a coset code which can detect $n - k - 1$ or less bits of slip and $d - 1$ or less additive errors if slip and additive error cannot occur simultaneously in any received n-tuple. However, the decoder cannot determine the nature of the error.

This theorem was quite easy to establish, but it does have some value. It demonstrates that by using coset codes it is possible to obtain maximum slip detection (for coset codes) without losing any additive-error-detecting ability. In addition the simplicity of using cyclic codes is retained.

B. Correction

In order to correct either (or both) type of error, the syndromes for slip must be different from the syndromes for additive error. In the next theorem, namely, Theorem

3.2A , both types of additive error are corrected. Theorem 3.2A is an improvement on Tong's[58] Theorem 4 in two senses. First, it will be shown here that the decoder can determine the magnitude as well as the direction of the slip. The second is that a redundant condition stated by Tong has been removed.

Let d be the minimum distance of all the cyclic codes (and hence their coset codes) considered in this chapter. Note also that for all the theorems in this chapter the phrase "correction of slip" will imply that the decoder can determine both the magnitude and the direction of the slip.

Theorem 3.2A

Given any (n , k) cyclic code, there exists an (n , k) coset code which can correct both e or less additive errors and S or less bits of slip, if additive error and slip do not occur simultaneously in any received n - tuple, where *

$$S = \text{Min.} \left\{ d - 2(e+1) , \left[\frac{n - k - e + [e/2] - 1}{[e/2] + 2} \right] \right\} .$$

Proof : The proof consists of two parts. The first part will show that the syndromes for all slip $\leq S$ are different from the syndromes for e or less additive errors. The second part will show that the syndromes for right slip and left slip are distinct for all slip $\leq S$.

---

* $[X]$ denotes the integer part of X .

It will also be pointed out that the decoder can determine the magnitude and the direction of the slip, for all slip $\leq S$. A procedure is also given for constructing a suitable $C(x)$.[58]

Assume that the transmitted code word $W_1(x) + C(x)$ suffers s bits of left slip and that the word $W_2(x) + C(x)$ has additive errors $E(x)$. Then, it is sufficient to show that

$$\left\{x^s (W_1(x) + C(x)) - C(x) + U_s(x)\right\} \neq \left\{W_2(x) + E(x)\right\} \tag{3.1}$$

for all $U_s(x)$, $s \leq S$ and $W\left[E(x)\right] \leq e$. The above relation can be simplified as

$$\left\{C(x)(x^s - 1) + U_s(x) + E'(x)\right\} \neq 0 \tag{3.2}$$

where $E'(x) = -E(x)$ and $E'(x)$ has the same weight as $E(x)$. However, the prime will be dropped in the next expression and $E(x)$ will always be written with a positive sign in the future. Since the minimum weight of a non-zero code word is d, (3.2) is satisfied if

$$0 < W\left[C(x)(x^s - 1) + U_s(x) + E(x)\right] < d \tag{3.3}$$

for all $U_s(x)$, $s \leq S$ and $W\left[E(x)\right] \leq e$. Let $D_s(x)$ be a polynomial derived from $C(x)(x^s - 1)$ by setting all the terms of degree $(s-1)$ or less to zero. Then, (3.3) can be written as

$$0 < W\left[D_s(x) + U_s(x) + E(x)\right] < d \tag{3.4}$$

since $U_s(x)$ is a polynomial with random coefficients. Notice that $U_s(x)$ cannot affect

the terms in $D_s(x)$ since $U_s(x)$ has no terms of degree s or greater. The left inequality in (3.4) is satisfied if and only if

$$W\left[D_s(x)\right] > e \qquad \text{for all } s \leq S \tag{3.5}$$

and the right inequality is satisfied if and only if

$$W\left[D_s(x)\right] + S + e < d \qquad \text{for all } s \leq S\,. \tag{3.6}$$

For a given e, it is desirable to maximize the correctable slip S. This requires that $W\left[D_s(x)\right]$ in (3.6) be a minimum, hence, to satisfy (3.5), assume that $D_s(x)$ can be chosen so that

$$W\left[D_s(x)\right] = e + 1 \quad \text{for all } s \leq S\,. \tag{3.7}$$

Substituting (3.7) in (3.6), gives

$$S < d - 2e - 1$$

or

$$S \leq d - 2(e + 1)\,. \tag{3.8}$$

Thus if a suitable $C(x)$ can be found, the decoder can distinguish between left slip and additive error. A similar result can be obtained for right slip.

In the second part of the proof it will be shown that the decoder can distinguish between left and right slip. This can be achieved if and only if (2.12) is satisfied, which

reduces to

$$\left\{C(x)\,(x^{s+r}-1) + U_{s+r}(x)\right\} \neq 0 \,. \qquad (2.15)$$

A choice of $C(x)$ that satisfies (3.8) and (2.15) is now described * . Let

$$C(x) = \sum_{i=o}^{e/2} x^{i(S+1)} \quad \text{for } e \text{ even} \qquad (3.9a)$$

and

$$C(x) = \sum_{i=o}^{[e/2]} x^{i(S+1)} + x^{n-1} \quad \text{for } e \text{ odd} \,. \qquad (3.9b)$$

Written out as n - tuples, (3.9a) and (3.9b) have the form (3.10a) and (3.10b), respectively, as shown

$$\underbrace{(10\ldots0,\ 10\ldots0,\ \ldots,\ \ldots,\ 10\ldots0,\ 0\ldots0)}_{(e/2\,+\,1)\,(S+1)} \qquad (3.10a)$$

($S+1$ zeros per block)

$$\underbrace{(10\ldots0,\ 10\ldots0,\ \ldots,\ \ldots,\ 10\ldots0,\ 0\ldots01)}_{(\,[e/2]\,+\,1)\,(S+1)} \qquad (3.10b)$$

($S+1$ zeros per block)

The motive for the choices of $C(x)$ in the above can easily be explained. First, it can be verified that $C(x)$ satisfies (3.7) for both even and odd e. Second, $C(x)$ is chosen

---

* These polynomials are very similar to the ones described in Tong.[58]

to obtain the largest value of slip which satisfies (2.15) * .

Substituting (3.9a) into (2.15) , the highest term, for e even, has degree $\frac{e}{2}(S+1) + (s+r)$. Recall that a burst of length $n-k$ or less cannot be an (n, k) cyclic code word. Hence, if

$$\frac{e}{2}(S+1) + (s+r) \leq n-k-1 \tag{3.11}$$

(2.15) will be satisfied. The maximum value of both s and r is S and hence (3.11) becomes

$$S\left(e/2 + 2\right) \leq n-k-\frac{e}{2} - 1$$

or

$$S \leq \frac{n-k-e/2-1}{e/2+2} \quad \text{for e even .} \tag{3.12a}$$

When e is odd, (3.9b) is substituted into (2.15) . The highest term will then be x^{n-1}, but by multiplying through by x , x^{n-1} becomes $x^n = 1$ and the new highest term has degree $[e/2]\,(S+1) + (s+r) + 1$. Hence, the polynomial in (2.15) is not a cyclic code word if

$$[e/2]\,(S+1) + (s+r) + 1 \leq n-k-1 \ .$$

---

* However, it is not being claimed that there do not exist other choices of C (x) which can improve the performance of Theorem 3.2A .

Replacing s and r by their maximum value S , and solving for S , the above becomes :

$$S \leq \frac{n - k - [e/2] - 2}{[e/2] + 2} \qquad \text{for } e \text{ odd .} \tag{3.12b}$$

The expressions for even and odd cases of e can be combined to give

$$S \left[\frac{n - k - e + [e/2] - i}{[e/2] + 2} \right] . \tag{3.12}$$

Using arguments similar to the ones above, it can further be shown that the syndromes for two left slips s_1 and s_2 , $s_1 \neq s_2$, are distinct for all $s_1 , s_2 \leq S$, and similarly for two right slips. The above implies that the decoder can compute both the magnitude and the direction of the slip, for all slip $\leq S$.

To complete the proof, it is necessary to consider if there is a minimum value of n that guarantees the existence of C (x) and at the same time ensures that $W [D_s (x)] = e + 1$, for all $s \leq S$. The latter condition requires that there are at least S zeros beyond the highest term in C (x) , excluding the term x^{n-1} , as illustrated in (3.10a) and (3.10b). Hence, a satisfactory C (x) exists if

$$n \geq (S + 1) (e/2 + 1) \qquad \text{for } e \text{ even} \tag{3.13a}$$

and

$$n \geq (S + 1) ([e/2] + 1) + 1 \qquad \text{for } e \text{ odd .} \tag{3.13b}$$

It is not difficult to show that (3.12a) implies (3.13a) and (3.12b) implies (3.13b) .

Finally, since S must satisfy (3.8) and (3.12), it is chosen as the minimum of the two upper bounds.

Q.E.D.

The next theorem is similar to Theorem 3.2A, and for which it is also assumed that slip and additive error do not occur simultaneously. One important difference, however, is that Theorem 3.2B is valid only for binary cyclic codes.

Theorem 3.2B

Any (n , k) binary cyclic code has a coset code which can correct both e or less additive errors and S or less bits of slip, if additive error and slip do not occur simultaneously in any received n - tuple, where

$$S = \text{Min.} \left\{ \left[\frac{3d-5}{2} - 3e - 2\left[e/2\right] \right], \left[\frac{n-k-e+\left[e/2\right]-1}{\left[e/2\right]+2} \right] \right\} .$$

Proof : Choose C (x) to be the same as in Theorem 3.2A . Then, the decoder can distinguish between left slip and additive error if and only if (3.1) in the proof of Theorem 3.2A is satisfied, and (3.1) can be reduced to

$$\left\{ D_s(x) + U_s(x) + E(x) \right\} \neq 0 , \qquad \text{all } s \leq S \tag{3.14}$$

where $D_s(x)$ was defind in the proof of Theorem 3.2A. Since $W\left[D_s(x)\right] = e + 1$ for all $s \leq S$, and $W\left[E(x)\right] \leq e$, the maximum weight of the polynomial in (3.14) is

$$e + 1 + S + e .$$

However, suppose that

$$W\left[U_s(x) + E(x)\right] = s + e - m \tag{3.15}$$

then (3.14) is satisfied if

$$(S + 2e + 1) - m < d . \tag{3.16}$$

Now, let

$$Q(x) \triangleq D_s(x) + U_s(x) + E(x)$$

and compute $x\,Q(x) + Q(x)$. If there are m terms missing from $U_s(x) + E(x)$, i.e., if (3.15) is true, it can be shown in a straightforward manner that the maximum weight of $Q(x)(x+1)$ is

$$2m + 2 + 2e + 2e \quad , \ e \text{ even},$$

where the above statement is valid only for binary codes. Note that the maximum is achieved when all of the m missing terms come from $U_s(x)$.

If $Q(x)(x+1)$ is not a cyclic code word, neither is $Q(x)$, hence the polynomial in (3.14) is not a cyclic code word if

$$2m + 4e + 2 < d \quad , \ e \text{ even}. \tag{3.17}$$

If either (3.16) or (3.17) is true, (3.14) is satisfied, and eliminating m from (3.16) and (3.17) gives

$$S < \frac{3d}{2} - (4e + 2) \quad \text{for } e \text{ even} \tag{3.18}$$

and for e odd,

$$S < \frac{3d}{2} - (4e + 1) \quad \text{for } e \text{ odd} . \tag{3.19}$$

Combining the even and odd cases give

$$S < \frac{3d}{2} - (4e + 2) + e - 2\left[e/2\right] \tag{3.20}$$

or

$$S \leqslant \frac{3d - 5}{2} - 3e - 2\left[e/2\right] . \tag{3.21}$$

A similar proof holds for right slip.

It remains to be shown that the decoder can determine the magnitude and the direction of the slip for all slip $\leqslant S$. This part of the proof is essentially identical to the second part of the proof of Theorem 3.2A and will not be repeated here. The result is that

$$S \leqslant \left[\frac{n - k - e + \left[e/2\right] - 1}{\left[e/2\right] + 2}\right] . \tag{3.22}$$

S is then chosen to be the minimum of (3.21) and (3.22).

Q.E.D.

If Theorems 3.2A and 3.2B are compared, their relative merits are fairly obvious. In general, Theorem 3.2B will give a larger value of S when d is large and e is small.

However, in any particular situation, it is fairly simple to calculate S by both theorems and choose the one which gives the larger S.

Actually, Theorem 3.2B can be improved upon by calculating the maximum weight of $Q(x)(x^i + 1)$, where $i = S - s + 1$, rather than $Q(x)(x + 1)$. Assuming that

$$W\left[U_s(x) + E(x)\right] = s + e - m \tag{3.23}$$

where $o \leq m \leq p + e$, then $Q(x)$ is not a cyclic code word if

$$2e + 1 + s - m < d. \tag{3.24}$$

Now, if $Q(x)(x^i + 1)$ is not a cyclic code word, then neither is $Q(x)$. It is shown in Appendix I that if (3.23) is satisfied, then

$$W\left[Q(x)(x^i + 1)\right] \leq 2 + 3e + 2S - 2(s - m). \tag{3.25}$$

It follows that $Q(x)$ can not be a cyclic code word if

$$2 + 3e + 2S - 2(s - m) < d \quad \text{for } e \text{ even}. \tag{3.26}$$

If either (3.24) or (3.26) is satisfied, then (3.14) is satisfied, and eliminating $(s - m)$ from (3.24) and (3.26) gives

$$S \leq (3d - 7e - 5)/2 \quad \text{for } e \text{ even}. \tag{3.27}$$

It can also be shown that

$$S \leqslant (3d - 7e - 4)/2 \qquad \text{for } e \text{ odd} \tag{3.28}$$

and (3.27) and (3.28) can be replaced by

$$S \leqslant \frac{3d-5}{2} - 3e - [e/2] \quad . \tag{3.29}$$

The above discussion is summarized in the next theorem.

Theorem 3.2C

Any (n , k) binary cyclic code has a coset code which can correct both e or less additive errors and S or less bits of slip, if additive error and slip do not occur simultaneously in any received n - tuple, where

$$S = \text{Min.} \left\{ \left[\frac{3d-5}{2} - 3e - [e/2] \right], \left[\frac{n-k-e+[e/2]-1}{[e/2]+2} \right] \right\}.$$

3.3. Case II : Slip And Additive Error Can Occur Simultaneously

As in Case I, Case II may be divided into two parts : A , detection and B , correction.

A. Detection

Here, the decoder will not attempt to distinguish between the two types of error or to correct either of them but will merely detect that some kind of error is present.

The next theorem, Theorem 3.3A , is valid for cyclic codes over any finite field F. Let d be the minimum distance for all the cyclic codes considered.

Theorem 3.3A

Any (n , k) cyclic code has a coset code which can detect the simultaneous occurrence of e or less additive errors and S or less bits of slip, where

$$S = \text{Min.} \left\{ d - 2(e+1) , \left[\frac{n - e + [e/2] - 1}{[e/2] + 1} \right] \right\} ,$$

but the decoder cannot determine the nature of the error.

Proof : Let W (x) be a cyclic code word, and assume that additive error E (x) and s bits of left slip have occurred. It is sufficient to show that

$$\left\{ x^s (W(x) + C(x)) - C(x) + U_s(x) + E(x) \right\} \neq 0 \qquad (3.30)$$

for all $s \leq S$ and $W[E(x)] \leq e$. Since $\left\{ x^s W(x) \right\} = 0$, (3.30) simplifies to

$$\left\{ C(x)(x^s - 1) + U_s(x) + E(x) \right\} \neq 0 . \qquad (3.31)$$

If (3.31) is compared with (3.2) , it is seen that they are identical. Choose C (x) to be the same as in Theorem 3.2A. Hence (3.31) is satisfied if $S \leq d - 2(e+1)$ as found in (3.8) . Also, (3.13a) is a necessary condition when e is even and (3.13b) is necessary when e is odd . Combining the even and odd cases and solving for S gives

$$S \leq \left[\frac{n - e + [e/2] - 1}{[e/2] + 1} \right] .$$

Since S must satisfy both of the bounds given above, it is chosen as the minimum of the two. The theorem was proved for left slip, but a similar proof applied for right slip.

Q.E.D.

Observe that if $s = 0$, (3.30) becomes

$$\left\{W(x) + C(x) - C(x) + E(x)\right\} \neq 0$$

which simplifies to give

$$\left\{E(x)\right\} \neq 0 \quad .$$

This implies that if no slip has occurred, the decoder can detect $d - 1$ or less additive errors. However, as mentioned in Theorem 3.3A, the decoder cannot determine whether the error was due to slip, additive error or both. Just as Theorem 3.2B improves on Theorem 3.2A for the special case of binary codes, the next theorem also improves on Theorem 3.3A but is valid only for binary codes. Theorem 3.3B is preferable when d is large and e is small.

Theorem 3.3B

Any (n, k) binary cyclic code has a coset code which can detect the simultaneous occurrence of e or less additive errors and S or less bits of slip, where

$$S = \text{Min.}\left\{\left[\frac{3d-5}{2} - 3e - 2\left[e/2\right]\right], \left[\frac{n-e+\left[e/2\right]-1}{\left[e/2\right]+1}\right]\right\},$$

but the decoder cannot determine the nature of the error.

Proof : Choose C (x) to be the same as in Theorem 3.2A . Let W (x) be any (n , k) binary cyclic code word, and assume that additive error E (x) and s bits of left slip have occurred. The decoder will detect the presence of error if and only if

$$\left\{ x^s (W(x) + C(x)) + U_s(x) + E(x) - C(x) \right\} \neq 0 \qquad (3.32)$$

which reduces to

$$\left\{ C(x)(x^s - 1) + U_s(x) + E(x) \right\} \neq 0. \qquad (3.33)$$

Recalling the definition of $D_s(x)$ in the proof of Theorem 3.2A , (3.33) can be written as

$$\left\{ D_s(x) + U_s(x) + E(x) \right\} \neq 0. \qquad (3.34)$$

Now, (3.34) is identical to (3.14) in the proof of Theorem 3.2B, and the same procedure can be followed to show that when

$$S \leq \frac{3d - 5}{2} - 3e - 2\left[e/2\right] \qquad (3.35)$$

then (3.34) is satisfied. Also, since C (x) is the same as in Theorems 3.2A and 3.3A, (3.13a) is a necessary condition when e is even, and (3.13b) is necessary when e is odd. Combining the even and odd cases and solving for S gives

$$S \left[\frac{n - e + \left[e/2\right] - 1}{\left[e/2\right] + 1} \right] . \qquad (3.36)$$

S is now chosen as the minimum of (3.35) and (3.36).

Q.E.D.

There is also a theorem which improves slightly on Theorem 3.3B and which is given below as Theorem 3.3C . The proof is omitted since it is readily obtained from the proofs of Theorem 3.2C and Theorem 3.3A.

Theorem 3.3C

Any (n , k) binary cyclic code has a coset code which can detect the simultaneous occurrence of e or less additive errors and S or less bits of slip, where

$$S = \text{Min.} \left\{ \left[\frac{3d-5}{2} - 3e - \left[e/2 \right] \right], \left[\frac{n - e + \left[e/2 \right] - 1}{\left[e/2 \right] + 1} \right] \right\},$$

but the decoder cannot determine the nature of the error.

B. Correction

We will first consider a decoder which can correct additive errors when no slip is present and can detect slip even when additive errors occur simultaneously.

Theorem 3.4

Any (n , k) cyclic code has a coset code which can correct e or less additive errors when no slip is present, and can also detect the simultaneous occurrence of S or less bits of slip and e or less additive errors, where

$$S = \text{Min.} \left\{ d - 2(2e+1) , \left[\frac{n-e-1}{e+1} \right] \right\}.$$

Proof : It is sufficient to show that the syndromes for slip plus additive error are different from the syndromes for additive error alone. Hence, assuming left slip of s bits, it is sufficient to show that

$$\left\{x^s (W_1 (x) + C (x)) - C (x) + U_s (x) + E_1 (x)\right\} \neq \left\{W_2 (x) + E_2 (x)\right\} \quad (3.37)$$

for all $s \leq S$, $W\left[E_i (x)\right] \leq e$, $i = 1, 2$, and $W_1 (x)$ and $W_2 (x)$ are any two cyclic code words. Recalling that the syndrome of a cyclic code word is zero and putting all terms of (3.37) on the left, we have

$$\left\{C (x) (x^s - 1) + U_s (x) + E_1 (x) + E_2 (x)\right\} \neq 0 \quad (3.38)$$

where, by convention, the sign of the additive error terms is always positive. Comparing (3.38) with (3.2) it is seen that the difference is that $E_1 (x) + E_2 (x)$ replaces $E (x)$ in (3.2). Making this substitution, a sufficient condition to satisfy (3.38) is again provided by (3.3). Hence from (3.5) it is required that

$$W\left[D_s (x)\right] > 2e \quad \text{for all} \quad s \leq S$$

and S will be a maximum if the choice of $C (x)$ gives

$$W\left[D_s (x)\right] = 2e + 1 \quad \text{for all} \quad s \leq S .$$

Let

$$C (x) = \sum_{i=o}^{e} x^{i (S+1)} .$$

Then it is easy to see that $W\left[C (x)\right] = e + 1$, and that $W\left[C (x) (x^s - 1)\right] = 2e + 2$

for all $s \leq S$. It follows that $W\left[D_s(x)\right] = 2e+1$ as desired. Now, from (3.6), replacing e in (3.6) by $2e$, we have

$$(2e+1) + S + 2e < d$$

or

$$S \leq d - 2(2e+1) \quad . \tag{3.39}$$

To ensure that $W\left[D_s(x)\right] = 2e+1$ for all $s \leq S$, it is necessary that there exists at least S zeros beyond the highest term in $C(x)$. In other words, it is necessary that

$$n \geq 1 + e(S+1) + S$$

or

$$S \leq \frac{n-e-1}{e+1} \quad . \tag{3.40}$$

S is then chosen as the minimum of (3.39) and (3.40). A similar proof applies for right slip.

Q.E.D.

Note that it is the total weight $W\left[E_1(x) + E_2(x)\right]$ that is significant and not their individual weights. Hence, the total weight can be distributed between $E_1(x)$ and $E_2(x)$ as desired. This fact is stated explicitly in the following corollary.

Corollary 3.1

Any (n, k) cyclic code has a coset code which can correct e_1 or less additive errors when no slip is present and which can also detect the simultaneous occur-

rence of e_2 or less additive errors and S or less bits of slip, where

$$S = \text{Min.} \left\{ d - 2(e+1), \left[\frac{n - e + [e/2] - 1}{[e/2] + 1} \right] \right\} ,$$

and $e \triangleq e_1 + e_2$.

Proof : In Theorem 3.4 , the maximum weight of $E_1(x) + E_2(x)$ was $2e$ which is always even. However, in Corollary 3.1 , $e = e_1 + e_2$ can be even or odd and a different $C(x)$ is required for each case. When e is even, the choice of $C(x)$ and the proof is identical to the proof of Theorem 3.4 . When e is odd, $C(x)$ is chosen as in (3.9b) , where in this case, $e = e_1 + e_2$ and (3.13b) is the required inequality, namely,

$$n \geq (S+1)([e/2] + 1) + 1 \quad , \quad e \text{ odd} .$$

Combining the even and odd cases give

$$S \leq \left[\frac{n - e + [e/2] - 1}{[e/2] + 1} \right] .$$

Corollary 3.1 then follows .

Q.E.D.

The ability of a coset code to correct the simultaneous occurrence of slip and additive error is now considered. In the next theorem, the decoder determines only the direction of the slip.

Theorem 3.5

Any (n, k) cyclic code has a coset code which can correct e or less additive errors and also determine the direction of the slip for S or less bits of slip, even when both errors occur simultaneously, where

$$S = \text{Min.} \left\{ \left[\frac{d - 2(2e+1)}{2} \right], \left[\frac{n - e - 1}{2(e+1)} \right] \right\}.$$

Proof : Choose $C(x) = \sum_{i=o}^{e} x^{i(2S+1)}$.

It is sufficient to show that the syndromes for left slip plus additive error are different from the syndromes for right slip plus additive error, i.e.,

$$\left\{ x^{s} (W_1(x) + C(x)) - C(x) + U_s(x) + E_1(x) \right\}$$

$$\neq \left\{ x^{-r} (W_2(x) + C(x)) - C(x) + U_{-r}(x) + E_2(x) \right\} \qquad (3.41)$$

for all $r, s \leq S$ and $W[E_i(x)] \leq e$, $i = 1, 2$, where $W_1(x)$ and $W_2(x)$ are any two cyclic code words. Putting all terms on the left, (3.41) can be simplified as

$$\left\{ C(x)(x^{s} - x^{-r}) + U_s(x) + U_{-r}(x) + E_1(x) + E_2(x) \right\} \neq 0 . \qquad (3.42)$$

Now, (3.42) is satisfied if and only if

$$\left\{ C(x)(x^{s+r} - 1) + x^{r} U_s(x) + U_r(x) + x^{r} E_1(x) + E_2(x) \right\} \neq 0 . \qquad (3.43)$$

where (3.43) is obtained from (3.42) by performing a cyclic shift of r bits (by multiply-

ing through by x^r) . Making the substitution $U_{s+r}(x) = x^r U_s(x) + U_r(x)$ and replacing $x^r E_1(x) + E_2(x)$ by $E(x)$, (3.43) becomes

$$\left\{C(x)(x^{s+r} - 1) + U_{s+r}(x) + E(x)\right\} \neq 0 \qquad (3.44)$$

where $W\left[E(x)\right] \leq 2e$. Let $D_{s+r}(x)$ be equal to $C(x)(x^{s+r} - 1)$ minus the terms of degree $s + r - 1$ or less, then (3.44) can be written as

$$\left\{D_{s+r}(x) + U_{s+r}(x) + E(x)\right\} \neq 0 \ . \qquad (3.45)$$

Since the minimum weight of a non-zero code word is d, (3.45) is satisfied if

$$0 < W\left[D_{s+r}(x) + U_{s+r}(x) + E(x)\right] < d \ . \qquad (3.46)$$

The lefthand inequality in (3.46) is satisfied for all $s, r \leq S$ and $W\left[E(x)\right] \leq 2e$ if and only if

$$W\left[D_{s+r}(x)\right] > 2e \ ,$$

and the righthand inequality is satisfied if and only if

$$W\left[D_{s+r}(x)\right] + 2S + 2e < d \ . \qquad (3.47)$$

The maximum value for S is obtained by letting

$$W\left[D_{s+r}(x)\right] = 2e + 1 \quad , \quad \text{all } s, r \leq S \ . \qquad (3.48)$$

It can be verified that the choice of $C(x)$ given for this proof satisfies (3.48). Substi-

tuting (3.48) into (3.47) gives

$$2e + 1 + 2S + 2e < d$$

which implies, since S is an integer,

$$S \leq \left[\frac{d - 2(2e + 1)}{2}\right] . \tag{3.49}$$

To satisfy (3.48) it is necessary that there exists 2 S zeros beyond the highest term in C (x). This is true if

$$n \geq 1 + e(2S + 1) + 2S$$

and solving for S gives

$$S \leq \left[\frac{n - e - 1}{2(e + 1)}\right] . \tag{3.50}$$

S is then chosen to be the minimum of (3.49) and (3.50).

Q.E.D.

In Theorem 3.5, it was proved that the decoder can distinguish between left slip plus additive error and right slip plus additive error. However, in order to recover sync on the next code word, it is also necessary to prove that the syndromes for any two left slips $s_1 \neq s_2$, $s_i \leq S$, $i = 1, 2$, plus additive errors are distinct, and similarly for any two right slips $r_1 \neq r_2$, plus additive error.

It will now be shown that the coset code defined in Theorem 3.5 can distinguish between two left slips plus additive error, but not between two right slips plus additive

error. Let $W_1(x)$ and $W_2(x)$ be any two cyclic code words, and assume that q and p bits of left slip and $E_1(x)$ and $E_2(x)$ additive errors occur to each, respectively. It is sufficient to show that, assuming $p > q$,

$$\left\{x^q (W_1(x) + C(x)) + U_q(x) + E_1(x)\right\} \neq \left\{x^p (W_2(x) + C(x)) + U_p(x) + E_2(x)\right\} \quad (3.51)$$

which reduces to

$$\left\{C(x)(x^p - x^q) + U_p(x) + U_q(x) + E_1(x) + E_2(x)\right\} \neq 0 \quad (3.52)$$

where $q < p \leq S$ and $W\left[E_i(x)\right] \leq e$. Letting $E(x) = E_1(x) + E_2(x)$ and $U'_p(x) = U_p(x) + U_q(x)$, (3.52) becomes

$$\left\{C(x)(x^p - x^q) + U'_p(x) + E(x)\right\} \neq 0 . \quad (3.53)$$

Now, $W\left[C(x)\right] = e + 1$ and since $q < p \leq S$, none of the terms in $C(x)(x^p - x^q)$ will cancel each other and hence $W\left[C(x)(x^p - x^q)\right] = 2e + 2$. $U'_p(x)$ may cancel one term and $E(x)$ may cancel $2e$ terms and hence there is at least one term left in the polynomial in (3.53). This proves that the polynomial in (3.53) cannot be the zero word and because of (3.46) it cannot be any other cyclic code word, hence (3.53) is satisfied. If right slip was assumed instead, it would be found that the polynomial corresponding to the one in (3.53) could become zero and the inequality would be violated. This means that the coset code given by $C(x) = \sum_{i=o}^{e} x^{i(2S+1)}$ cannot distinguish between two right slips plus additive error. This can be remedied by choosing $C(x) = x^{n-1} + \sum_{i=o}^{e} x^{i(2S+1)}$, which reduces the value of S in Theorem 3.5 by at most one. For this $C(x)$, (3.49)

now becomes

$$S \leq \left[\frac{d - 4e - 3}{2}\right] \tag{3.54}$$

and (3.27) becomes

$$S \leq \left[\frac{n - e - 2}{2(e + 1)}\right] . \tag{3.55}$$

The above discussion may be summarized in the next theorem, where the phrase "correction of slip" means that the decoder can determine the magnitude as well as the direction of the slip.

Theorem 3.6

Given an (n, k) cyclic code, there exists an (n, k) coset code which can simultaneously correct e or less additive errors and S or less bits of slip, where

$$S = \text{Min.} \left\{ \left[\frac{d - 4e - 3}{2}\right], \left[\frac{n - e - 2}{2(e + 1)}\right] \right\} .$$

An upper bound on the ability of cyclic codes to correct the simultaneous occurrence of slip and additive error is readily obtained by rewriting (3.44) as shown below

$$\left\{C(x)(x^{s+r} - 1)\right\} \neq \left\{U_{s+r}(x) + E(x)\right\} , \tag{3.56}$$

for all $U_{s+r}(x)$, $s, r \leq S$ and all $E(x)$ such that $W\left[E(x)\right] \leq 2e$. However, when

$s + r + 2e \geq n - k$ and $s + r \leq 2k + 2e$, the random polynomial $U_{s+r}(x) + E(x)$ can generate all possible syndromes, and hence the inequality (3.56) cannot be guaranteed. The condition $s + r \leq 2k + 2e$ is imposed to ensure that the coefficients of $U_{s+r}(x)$ are completely arbitrary. The above discussion establishes the following theorem :

Theorem 3.7

No coset of any (n, k) cyclic code can simultaneously correct e additive errors and s bits of slip, when the slip lies in the range

$$(n - k - 2e)/2 \leq s \leq k + e .$$

In Table III - 1, the performance of the coset codes of a few binary cyclic codes in detecting and correcting slip in a noisy channel is listed. Observe that for small values of e, Theorems 3.2B and 3.2 C correct larger values of slip, S, than Theorem 3.2A. Theorem 3.2A is essentially Tong's theorem[54], whereas 3.2B and 3.2C are believed to be new. Similarly, Theorems 3.3B and 3.3C are improvements on Theorem 3.3A when the number of additive errors, e, is small. Theorem 3.3A is implied in Tong's work[54] but he does not state it explicitly. Note that Theorem 3.6 does not apply to the (127,22) code since in this case, $3k < n$. To the best of the author's knowledge, Theorems 3.3B, 3.3C, 3.4, 3.5, 3.6, 3.7 and Corollary 3.1 are original. Theorem 3.1 is a statement of a somewhat obvious result. Note that, whenever valid, the theorems have been proved for cyclic codes over arbitrary finite fields.

TABLE III - 1

A Few Examples To Illustrate The Ability Of Coset Codes Of Cyclic Codes To Detect Or Correct Slip In A Noisy Channel. S Is The Maximum Allowable Slip And e Is The Number Of Additive Errors, Which Are Assumed To Be Independently Distributed.

(n , k) CYCLIC CODE	d	e	MAXIMUM SLIP S FOR THEOREMS LISTED									
			3.2A	3.2B	3.2C	3.3A	3.3B	3.3C	3.4	3.5	3.6	3.7 *
(23, 12)	7	1	3	4	4	3	5	5	1	0	0	4
		2	1	0	1	1	0	1	0	0	0	3 †
		3	0	0	0	0	0	0	0	0	0	2 †
(127, 85)	13	1	9	14	14	9	14	14	7	3	3	19
		2	7	9	10	7	9	10	3	1	1	18
		3	5	6	7	5	6	7	0	0	0	17
		4	3	1	3	3	1	3	0	0	0	16
(255, 191)	17	1	13	20	20	13	20	20	11	5	5	30
		2	11	15	16	11	15	16	7	3	3	29
		3	9	12	13	9	12	13	3	1	1	28
		4	7	7	9	7	7	9	0	0	0	27
		5	5	4	6	5	4	6	0	0	0	26
		6	3	0	2	3	0	2	0	0	0	25
(255, 163)	25	1	21	32	32	21	32	32	19	9	9	44
		2	19	27	28	19	27	28	15	7	7	43
		3	17	24	25	17	24	25	11	5	5	42
		4	15	19	21	15	19	21	7	3	3	41
		5	13	16	18	13	16	18	3	1	1	40
		6	11	11	14	11	11	14	0	0	0	39
		7	9	8	11	9	8	11	0	0	0	38
(127, 22)	47	1	43	51	51	43	65	65	41	20	20	
		2	34	34	34	41	60	61	37	18	18	
		3	34	34	34	39	57	58	31	15	15	
		4	25	25	25	37	41	41	24	12	12	
		5	25	25	25	35	41	41	20	10	10	
		6	20	20	20	30	30	30	17	8	8	
		7	20	20	20	30	30	30	15	7	7	
		8	16	16	16	24	24	24	13	6	6	
		9	16	16	16	24	24	24	9	4	4	

* Obtained by setting $S = \left[(n - k - 2e - 1)/2\right]$.

† By making use of the fact that this is a perfect code, it can be shown that these values can be replaced by zero .

CHAPTER IV

MATRIX APPROACH TO SYNCHRONIZATION RECOVERY FOR BINARY CYCLIC CODES

4.1 Vector - Matrix Description Of Cyclic Code Words

In this chapter, the code words of an (n, k) cyclic code will be considered as n - vectors with elements from a finite field F. These vectors will sometimes be written horizontally (i.e., as row vectors), e.g.,

$$\bar{A} = [a_1, a_2, \ldots\ldots, a_n] , \quad a_i \in F .$$

The context should make the meaning clear. It seems more natural to write the first element of the vector $\bar{A}$ as a_1 rather than a_o, as in the case of a polynomial. This slight difference will cause no misunderstanding, however.

An (n, k) cyclic code can be defined as a k - dimensional subspace of the n - dimensional space of all n - tuples whose elements belong to F, plus the property that a cyclic shift of the components of any code word is also a code word. A cyclic code can be specified by its generator matrix. Usually,[39] this is a $k \times n$ matrix in which the rows are any k linearly independent code words, hence the matrix has rank k. Here, the generator matrix G will be defined as the transpose of the above matrix.[52] Thus G will be an $n \times k$ matrix whose columns are code words. Another important matrix associated with cyclic codes (linear codes in general) is the parity check matrix H. It is an $(n - k) \times n$ matrix of rank $(n - k)$ and G defines the null space of H, which implies

$$H\,G = 0. \tag{4.1}$$

Now, any cyclic code word, $\overline{W}$, may be written as

$$\overline{W} = G\,\overline{X} \tag{4.2}$$

where $\overline{W}$ is an n - vector and $\overline{X}$ is a k - vector. There are q^k possible $\overline{X}$ vectors and hence, as is well known,[39] q^k cyclic code words.

If $\overline{W}$ is a cyclic code word, then

$$H\,\overline{W} = 0 .$$

This follows from (4.2) and (4.1). In fact, it can be shown that an n - vector $\overline{W}$ is a cyclic code word if and only if $H\,\overline{W} = 0$. The syndrome[39] of an n - vector Z may be defined as the $(n - k)$ - vector obtained by computing $H\,\overline{Z}$. Of course, a cyclic code word has syndrome zero.

The correspondence between the generator matrix G and the generator polynomial $G(x)$ should be noted. The matrix H also corresponds to $H(x)$.

Before transmission, an n - vector $\overline{C}$ is added to $\overline{W}$, which gives

$$\overline{B} = G\,\overline{X} + \overline{C} \tag{4.3}$$

where $\overline{B}$ is the vector actually transmitted. The words $\overline{B}$ define a coset code.

4.2 Vector - Matrix Description Of Slip

Partition the n - vector $\overline{W}$ into two parts $\overline{W}_1$ and $\overline{W}_2$, where $\overline{W}_1$ has s elements and $\overline{W}_2$ has $n - s$ elements. Similarly, partition G into $\begin{bmatrix} G_1 \\ G_2 \end{bmatrix}$,

G is an $s \times k$ matrix and G_2 is an $(n-s) \times k$ matrix. Then, $\overline{W}$ can be written as

$$\overline{W} = \begin{bmatrix} \overline{W}_1 \\ \overline{W}_2 \end{bmatrix} = \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} \overline{X} . \tag{4.4}$$

Since $\overline{W}$ is a cyclic code word, any cyclic shift of the components of $\overline{W}$ is also a cyclic code word, hence

$$\begin{bmatrix} \overline{W}_2 \\ \overline{W}_1 \end{bmatrix} = \begin{bmatrix} G_2 \\ G_1 \end{bmatrix} \overline{X} \tag{4.5}$$

is a cyclic code word for all cyclic shifts s and for all code words $\overline{W}$. It is also true that any cyclic rotation of the rows of G will produce another matrix $\begin{bmatrix} G_2 \\ G_1 \end{bmatrix}$ which will also be a generator matrix for the same code. Thus, any cyclic code word may be written as shown in (4.5).

Now, assume that the n-vector $\overline{B} = G\overline{X} + \overline{C}$ is transmitted and there is a loss of synchronization at the receiver. In particular, suppose that s bits of left slip occurs, then the receiver frames the n-vector shown in Figure 4.1, where $\overline{A}$ and $\overline{D}$ are the left-adjacent and right-adjacent transmitted words, respectively. Mathematically, the framed n-vector may be written as

$$\begin{bmatrix} \overline{A}_2 \\ \overline{B}_1 \end{bmatrix}_L = \begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_L \begin{bmatrix} \overline{X}_2 \\ \overline{X} \end{bmatrix} + \begin{bmatrix} \overline{C}_2 \\ \overline{C}_1 \end{bmatrix}_L \tag{4.6}$$

where $\overline{A}_2$ has s elements and comes from $\overline{A}$, the adjacent word on the left, and $\overline{B}_1$

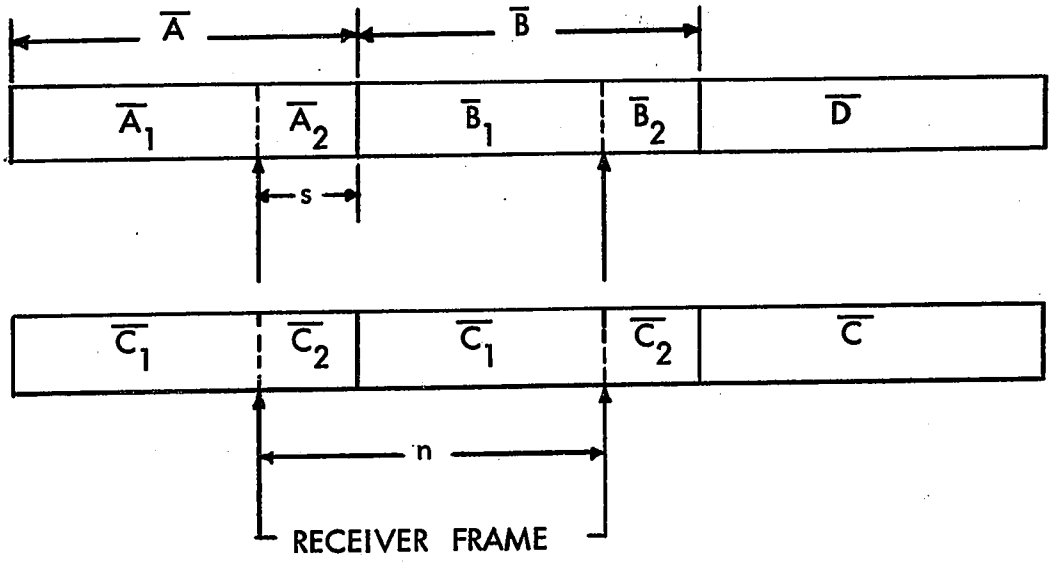

FIGURE 4.1 Receiver Frame for s Bits of Left Slip.

was originally the left - most $n - s$ elements of the transmitted word $\overline{B}$. In addition, the k - vectors $\overline{X}_2$ and $\overline{X}$ are defined by the relations

$$\overline{A} = G\overline{X}_2 + \overline{C} \tag{4.7a}$$

and

$$\overline{B} = G\overline{X} + \overline{C} \tag{4.7b}$$

The submatrix G_1 in (4.6) has $n - s$ rows and G_2 has s rows. From Figure 4.1, it is seen that $\begin{bmatrix} C_2 \\ C_1 \end{bmatrix}_L$ is obtained from $\overline{C}$ by cyclically shifting it s units to the left (or $n - s$ units to the right). Observe that (4.6) can be written as two vector equations, i. e.,

$$\bar{A}_2 = G_2 \bar{X}_2 + \bar{C}_2 \tag{4.8a}$$

and

$$\bar{B}_1 = G_1 \bar{X} + \bar{C}_1 \ . \tag{4.8b}$$

Similarly to (4.6), a right slip of r bits gives

$$\begin{bmatrix} \bar{B}_2 \\ \bar{D}_1 \end{bmatrix}_R = \begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_R \begin{bmatrix} \bar{X} \\ \bar{X}_1 \end{bmatrix} + \begin{bmatrix} \bar{C}_2 \\ \bar{C}_1 \end{bmatrix}_R \tag{4.9}$$

where $\bar{D}_1$ and $\bar{C}_1$ have r elements and the k - vector $\bar{X}_1$ is defined by

$$\bar{D} = G \bar{X}_1 + \bar{C}.$$

The subscripts L and R refer to left and right slip, respectively.

The above notation is explicit but somewhat cumbersome and an abbreviated notation will now be introduced. For (4.6), write

$$\bar{B}_L = G_L \bar{X} + \bar{C}_L \tag{4.10}$$

and for (4.9) write

$$\bar{B}_R = G_R \bar{X} + \bar{C}_R \ . \tag{4.11}$$

The amount of slip does not appear explicitly in the above expressions, but this omission should cause no misunderstanding.

In polynomial form, (4.10) and (4.11) could be written as

$$B_L(x) = x^s (W(x) + C(x)) + U_s(x)$$

and

$$B_R(x) = x^{-r} (W(x) + C(x)) + U_{-r}(x)$$

respectively.

4.3 Detection Of Slip For A Noiseless Channel

If slip is to be detected, the framed n - vector at the receiver must not be a transmitted word (coset code word). In decoding, the decoder first subtracts the vector $\overline{C}$ from the received word. If the difference is a cyclic code word, it is decided that no slip is present. Recall that an n - vector is a cyclic code word if and only if

$$H \overline{W} = 0. \tag{4.12}$$

Hence, from the above argument, the decoder concludes that slip is present if

$$H \left[\overline{B}_L - \overline{C} \right] \neq 0 \tag{4.13a}$$

for left slip (see (4.10) and

$$H \left[\overline{B}_R - \overline{C} \right] \neq 0 \tag{4.13b}$$

for right slip (see (4.11)) . In other words, the decoder decides that slip is present if the

syndrome of the received n - vector minus $\overline{C}$ is non zero. It should be noted that if the syndrome is zero, it does not necessarily imply that slip is absent. It could also mean that the method has failed to detect it. The problem then, essentially, is to find a vector $\overline{C}$ which will enable the decoder to detect (or correct) the widest range of slip possible.

Some of the theorems proved earlier will now be proved again, using the vector - matrix representation. For cross - reference, the theorems will be given corresponding numbers. Before proving the first theorem, a useful lemma[39] is stated and proved.

Lemma 4.1

A non-zero (n , k) cyclic code word cannot be a burst of length n - k or less.

Proof : By contradiction, assume that $\overline{W}$ is a cyclic code word which is a burst of length p, where $p \leq n - k$. Since the code is cyclic, $\overline{W}$ can be cyclically shifted until it has the form

$$\overline{W} = (\omega_1 , \omega_2 , \ldots \omega_{p-1} , \omega_p , 0 , \ldots 0) , \quad \omega_i \in F$$

and $\omega_1 , \omega_p \neq 0$, since $\overline{W}$ is a burst of length p . Now, form the matrix

$$G_p = \begin{bmatrix} \omega_1 , \omega_2 , \ldots , \omega_{p-1} , \omega_p , 0 , \ldots 0 \\ 0 , \omega_1 , \omega_2 , \ldots , \omega_{p-1} , \omega_p , \ldots 0 \\ \\ \ldots \ldots \ldots \ldots \ldots \\ \\ 0 , \ldots , 0 , \omega_1 , \omega_2 , \ldots , \omega_{p-1} , \omega_p \end{bmatrix} \quad n - p + 1$$

n

where the first row is $\overline{W}$ and the j'th row is obtained by cyclically shifting $\overline{W}$ by (j-1) bits as shown, until ω_p occupies the n'th column. This procedure generates an $(n - p + 1) \times n$ matrix whose rows are linearly independent cyclic code words. Hence they can be used to generate q^{n-p+1} distinct cyclic code words, and since $p \leq n - k$, it follows that $n - p + 1 \geq k + 1$. But this leads to a contradiction since there are q^k cyclic code words. Hence the assumption that a (non-zero) cyclic code word can be a burst of length $n - k$ or less in false.

Q.E.D.

Lemma 4.1 is stated as Theorem 8.2 in Peterson's book.[39] However, his proof is algebraic. The proof given here was developed by the author, who has not seen a similar one elsewhere.

Theorem 4.2.1

Any (n, k) cyclic code has at least one coset code which can detect all slip $\leq n - k - 1$ bits.

Proof : To generate a suitable coset code, add the n - vector $\overline{C} = [1\ 0\ \ldots\ 0]$ to each cyclic code word before transmission.* Assume that s bits of left slip occur at the receiver, which will detect the presence of slip if and only if

---

* The vector $\overline{C} = [0\ \ldots\ 0\ 1]$ will also give the same result. In fact, in both choices of $\overline{C}$, unity can be replaced by any element of the finite field F.

$$H\left[\bar{B}_L - \bar{C}\right] \neq 0$$

and substituting for $\bar{B}_L$ from (4.10) gives

$$H\left[G_L \bar{X} + \bar{C}_L - \bar{C}\right] \neq 0 \quad . \tag{4.14}$$

Any cyclic code word $\bar{W}$ may be written as

$$\bar{W} = \begin{bmatrix} G_2 \\ G_1 \end{bmatrix}_I \bar{Y} = \begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_I \begin{bmatrix} \bar{Y} \\ Y \end{bmatrix}, \quad I = L \text{ or } R, \tag{4.15}$$

where $\bar{Y}$ is a k - vector and is determined by the particular word under consideration. Observe that (4.15) is an identity. It is true that

$$\begin{aligned} G_L \bar{X} &\triangleq \begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_L \begin{bmatrix} \bar{X}_2 \\ \bar{X} \end{bmatrix} \\ &= \begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_L \begin{bmatrix} \bar{X} + \bar{X}'_2 \\ \bar{X} + 0 \end{bmatrix} \\ &= \begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_L \begin{bmatrix} \bar{X} \\ \bar{X} \end{bmatrix} + \begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_L \begin{bmatrix} \bar{X}'_2 \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} G_2 \\ G_1 \end{bmatrix}_L \bar{X} + \begin{bmatrix} G_2 \\ 0 \end{bmatrix}_L \bar{X}'_2 \end{aligned} \tag{4.16}$$

where, by definition, $\bar{X}_2 = \bar{X} + \bar{X}'_2$. Recalling the fact that $H\bar{W} = 0$, computing the syndrome of (4.16) gives

$$H G_L \bar{X} = H \begin{bmatrix} G_2 \\ 0 \end{bmatrix}_L \bar{X}'_2$$

where G_2 has s rows. Let

$$U_L \triangleq \begin{bmatrix} G_2 \\ 0 \end{bmatrix}_L \bar{X}'_2 \quad ,$$

then (4.14) may be written as

$$H \left[\bar{U}_L + \bar{C}_L - \bar{C} \right] \neq 0 \; . \tag{4.17}$$

$\bar{U}_L$ is the vector equivalent of the random polynomial $\bar{U}_s(x)$, where $\bar{U}_L$ is an n - vector whose first s terms are random and whose last n - s terms are zero. Hence it is a burst of length s or less. The vector $\bar{U}_L + \bar{C}_L - \bar{C}$ may be written as

$$\left[u'_1, u'_2, \ldots, u_s, 1, 0, \ldots, 0 \right], \; u_i \in F, \; u'_1 = u_1 - 1 \tag{4.18}$$

which is a burst of length s + 1 or less. But a burst of length n - k or less can not be an (n, k) cyclic code word, therefore (4.17) is satisfied if $s + 1 \leq n - k$, i.e., $s \leq n - k - 1$. If r bits of right slip was assumed, (4.17) would become

$$H \left[\bar{U}_R + \bar{C}_R - \bar{C} \right] \neq 0$$

and (4.18) would become

$$[-1, 0, \ldots, 0, v'_{n-r+1}, \ldots, v_{n-1}, v_n] \quad , v_i \in F \qquad (4.19)$$

where $v'_{n-r+1} = v_{n-r+1} + 1$, and (4.19) is also a burst of length $r+1$ or less. Hence, (4.19) is not a code word if $r \leq n-k-1$. It follows that the decoder can detect all slip less than or equal to $n-k-1$.

Q.E.D.

By comparing this proof with the proof of Theorem 2.1 , several similarities will be apparent, some of which have already been pointed out. In both cases the decoder subtracts a known n-tuple (C vs. C (x)) from the received n-tuple and decides that slip is present if the syndrome of the difference is non-zero. Also, a key point in both proofs is that an (n, k) cyclic code word cannot be a burst of length $n-k$ or less. However, the theorem can be proved without using this fact explicitly in either case[55,58], as was shown for Theorem 2.1 . It is difficult to compare the lengths of the proofs, as this depends largely on the amount of background material assumed. The author believes that many communication engineers are not familiar with modern algebra although nearly all have some knowledge of matrix theory. As a result, the matrix approach may seem simpler, largely because it is more familiar.

Before giving the next theorem, the following lemma is proved.

Lemma 4.2

Any (n, k) cyclic code with q symbols (or levels) has q^{n-k} distinct syndromes and they can all be generated by the random n-vector $\bar{V}$ which is arbitrary in a fixed interval of length $n-k$ and is zero elsewhere.

Proof : First, it is clear that $\overline{V}$ cannot be a non-zero (n, k) cyclic code word, since it is a burst of length $n-k$ or less. Hence, $H\,\overline{U} = 0$ if and only if $\overline{V} = 0$. Also, since $\overline{V}$ has $n-k$ arbitrary components, there are q^{n-k} distinct choices of $\overline{V}$. Now suppose, by contradiction, that two distinct choices of $\overline{U}$, $\overline{U}_i$ and $\overline{U}_j$, have identical syndromes, i. e.,

$$H\,\overline{U}_i = H\,\overline{U}_j$$

which implies

$$H\left[U_i - U_j\right] = 0$$

which in turn implies that $\overline{U}_i - \overline{U}_j$ is a cyclic code word. But $\overline{U}_i - \overline{U}_j$ is also a burst of length $n-k$ or less. Hence $H\left[\overline{U}_i - \overline{U}_j\right] = 0$ if and only if $\overline{U}_i - \overline{U}_j = 0$. But $\overline{U}_i$ and $\overline{U}_j$ are distinct and their differences cannot be zero. This proves that every distinct choice of $\overline{U}$ generates a distinct syndrome. The syndrome is an $n-k$ vector and so there are at most q^{n-k} distinct syndromes. It follows that there are q^{n-k} syndromes and that $\overline{U}$ can generate all of them.

Q.E.D.

It should be added that it is well known that an (n, k) cyclic code with q symbols has q^{n-k} distinct syndromes.

Theorem 4.2.2

No (n, k) cyclic code has a coset code which has comma - freedom exceeding $n-k-1$.

Proof : Allowing $\bar{C}$ to be arbitrary, the decoder can detect s bits of left slip if and only if (4.17) is satisfied, i.e.,

$$H\left[\bar{U}_L \pm \bar{C}_L - \bar{C}\right] \neq 0, \quad \text{for all } U_L \qquad (4.17)$$

and (4.17) can be written as

$$H\left[\bar{C}_L - \bar{C}\right] \neq H\,\bar{U}_L \qquad (4.20)$$

and since $\bar{U}_L$ has random coefficients, its sign will always be taken to be positive for convenience. When $s \geq n-k$, $\bar{U}_L$ will be a random burst of length $n-k$ or greater and this contains all random bursts of length $n-k$ or less. Hence by Lemma 4.2, when $s \geq n-k$, $H\,\bar{U}_L$ can be set equal to any desired syndrome. This means that regardless of the choice of $\bar{C}$, the inequality in (4.20) cannot be guaranteed for all $\bar{U}_L$ when $s \geq n-k$, i.e., when $s > n-k-1$. Similarly, for r bits of right slip, $\bar{U}_R$ can generate all syndromes when $r > n-k-1$. It follows that the comma - freedom cannot exceed $n-k-1$.

Q.E.D.

4.4 Correction Of Slip For A Noiseless Channel

In the next theorem, which is the same as Theorem 2.3, the decoder can distinguish between right slip and left slip.

Theorem 4.2.3

Any (n, k) cyclic code has at least one coset code which can distinguish between right slip and left slip if the slip does not exceed $(n - k - 1)/2$.

Proof: Let $\bar{C} = [1\,0\,\ldots\,0]$ and let $\bar{A}$ and $\bar{B}$ be any two transmitted code words where, $\bar{A} = G\bar{X}_r + \bar{C}$ and $\bar{B} = G\bar{X}_s + \bar{C}$. Also, assume that $\bar{A}$ slips to the left by s bits. Then, in the explicit matrix notation, it is sufficient to show that

$$H\left\{\begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_L \begin{bmatrix} \bar{X}_2 \\ X_s \end{bmatrix} + \begin{bmatrix} C_2 \\ C_1 \end{bmatrix}_L\right\} \neq H\left\{\begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_R \begin{bmatrix} \bar{X}_r \\ \bar{X}_1 \end{bmatrix} + \begin{bmatrix} \bar{C}_2 \\ \bar{C}_1 \end{bmatrix}_R\right\} \tag{4.21}$$

for all $r, s \leq (n - k - 1)/2$ and for all k-vectors $\bar{X}_1$ and $\bar{X}_2$, where $\bar{X}_1$ defines the word to the right of $\bar{A}$ and $\bar{X}_2$ defines the word to the left of $\bar{B}$. For left slip it has been shown that (see (4.15) and (4.16))

$$\begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_L \begin{bmatrix} \bar{X}_2 \\ \bar{X}_s \end{bmatrix} = \begin{bmatrix} G_2 \\ G_1 \end{bmatrix}_L \bar{X}_s + \begin{bmatrix} G_2 \\ 0 \end{bmatrix}_L \bar{X}'_2$$

where $\bar{X}_2 = \bar{X}_s + \bar{X}'_2$ and for right slip it is similarly true that

$$\begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_R \begin{bmatrix} \bar{X}_r \\ X_1 \end{bmatrix} = \begin{bmatrix} G_2 \\ G_1 \end{bmatrix}_R \bar{X}_r + \begin{bmatrix} 0 \\ G_1 \end{bmatrix}_R \bar{X}'_1$$

where $\bar{X}_1 = \bar{X}_r + \bar{X}'_1$. Note also that

$$\begin{bmatrix} G_2 \\ G_1 \end{bmatrix}_L \bar{X}_s \quad \text{and} \quad \begin{bmatrix} G_2 \\ G_1 \end{bmatrix} \bar{X}_r$$

are cyclic code words and hence their syndromes are zero. Applying the above results to (4.21) gives

$$H\left\{\begin{bmatrix} G_2 \\ 0 \end{bmatrix}_L \bar{X}'_2 + \begin{bmatrix} C_2 \\ C_1 \end{bmatrix}_L\right\} \neq H\left\{\begin{bmatrix} 0 \\ G_1 \end{bmatrix}_R \bar{X}'_1 + \begin{bmatrix} \bar{C}_2 \\ \bar{C}_1 \end{bmatrix}_R\right\} . \tag{4.22}$$

In the simplified matrix notation, (4.22) becomes

$$H\left[\bar{U}_L + \bar{C}_L\right] \neq H\left[\bar{V}_R + \bar{C}_R\right] \tag{4.23}$$

where $\bar{U}_L$ is a random burst which occupies the first s coordinates of the n-vector, and $\bar{V}_R$ is another random burst which lies in the last r coordinates. Putting all the terms of (4.23) on the left gives

$$H\left[\bar{U}_L + \bar{V}_R + \bar{C}_L - \bar{C}_R\right] \neq 0. \tag{4.24}$$

Recalling that $C = [1\ 0\ \ldots\ 0]$, the vector $\left[\bar{U}_L + \bar{V}_R + \bar{C}_L - \bar{C}_R\right]$ in (4.24) has the form

$$[v_1, v_2, \ldots, v_s, 1, 0, \ldots, 0, v'_{n-r+1}, \ldots, v_{n-1}, v_n] \tag{4.25}$$

where $v'_{n-r+1} = v_{n-r+1} - 1$ and (4.25) can be regarded as a burst of length at most $s + r + 1$ and hence (4.25) is not a cyclic code word if $s + r + 1 \leq n - k$, i. e., $s + r \leq n - k - 1$. Letting S be the maximum of both r and s, it can be concluded that the decoder can distinguish between left slip and right slip if $S \leq (n - k - 1)/2$.

Q.E.D.

As a reminder that the choice of coset is not unique, it is pointed out that choosing $\overline{C} = [0 \ . \ . \ . \ 0 \ 1]$ gives the same result in the above theorem.

4.5 Correction Of Slip For A Noisy Channel

Recall that Theorem 3.2A assumes that slip and additive error do not occur simultaneously. This theorem will be proved in this chapter because it introduces the factor of additive error. As before, let d be the minimum distance of the cyclic code.

Theorem 4.3.2A

Any (n , k) cyclic code, has a coset code which can correct e or less additive errors and S or less bits of slip, if additive error and slip do not occur simultaneously in any received n - tuple , where

$$S = \text{Min.} \left\{ d - 2(e + 1), \left[\frac{n - k - e + [e/2] - 1}{[e/2] + 2} \right] \right\}.$$

Proof : When e is even, let

$$\bar{C} = [1\overset{S+1}{0\ldots0}, 1\overset{S+1}{0\ldots0}, \ldots, \ldots, 1\overset{S+1}{0\ldots0}, \overset{}{0\ldots0}] \tag{4.26a}$$

where there are $(\frac{e}{2} + 1)$ blocks of length $S + 1$ on the left of $\bar{C}$ as shown, and $\bar{C}$ is zero elsewhere.

When e is odd, let

$$\bar{C} = [1\overset{S+1}{0\ldots0}, 1\overset{S+1}{0\ldots0}, \ldots, \ldots, 1\overset{S+1}{0\ldots0}, 0\ldots01] \tag{4.26b}$$

where there are $([\frac{e}{2}] + 1)$ blocks of length $S + 1$ on the left as shown and in addition there is a one at the extreme right, $\bar{C}$ is zero elsewhere. Now, (4.26a) will exist if

$$n \geq (\frac{e}{2} + 1)(S + 1) \tag{4.27a}$$

and (4.26b) will exist if

$$n \geq ([\frac{e}{2}] + 1)(S + 1) + 1 \ . \tag{4.27b}$$

The proof will consist essentially of two parts : the first part will show that the syndromes for slip are different from the syndromes for additive error, and the second part will show that the syndromes for right and left slip are distinct, for all slip $\leq S$.

Let $G\,\bar{X}_1$ and $G\,\bar{X}_2$ be any two cyclic code words, and also let $G\,\bar{X}_i$ be any cyclic code word to the left of $G\,\bar{X}_1$. Assuming left slip of s bits, the decoder can distinguish between slip and additive error, if and only if

$$H\left\{\left[\begin{matrix} G_2 & 0 \\ 0 & G_1 \end{matrix}\right]_L \left[\begin{matrix} \bar{X}_i \\ \bar{X}_1 \end{matrix}\right] + \left[\begin{matrix} \bar{C}_2 \\ \bar{C}_1 \end{matrix}\right]_L - \bar{C}\right\} \neq H\left[G\,\bar{X}_2 + \bar{E}\right] \qquad (4.28)$$

where $\bar{E}$ is the n - vector representing additive error. It is assumed the $W\left[\bar{E}\right] \leq e$ where $W\left[\bar{E}\right]$ means the weight of $\bar{E}$, and is defined as the number of non-zero terms in $\bar{E}$. By already developed methods, (4.28) can be simplified to

$$H\left\{\left[\begin{matrix} G_2 \\ 0 \end{matrix}\right]_L \bar{X}_1' + \left[\begin{matrix} \bar{C}_2 \\ C_1 \end{matrix}\right]_L - \bar{C} + \bar{E}\right\} \neq 0 \qquad (4.29)$$

Writing (4.29) in the shorter notation gives

$$H\left[\bar{U}_L + \bar{C}_L - \bar{C} + \bar{E}\right] \neq 0 \quad . \qquad (4.30)$$

Now, (4.30) is satisfied if

$$0 < W\left[\bar{U}_L + \bar{C}_L - \bar{C} + \bar{E}\right] < d \qquad (4.31)$$

for all $\bar{U}_L$, $s \leq S$, $W\left[\bar{E}\right] \leq e$. Let $\bar{D}_s$ be the n - vector obtained by setting the first s elements of $\bar{C}_L - \bar{C}$ equal to zero. Then, (4.31) can be written as

$$0 < W\left[\bar{D}_s + \bar{U}_L + \bar{E}\right] < d \; . \qquad (4.32)$$

From here on, the proof of the first part is essentially the same as before.

In the second part of the proof, it is shown that the syndromes for left and right slip are distinct. This is true if and only if (4.21) is satisfied for $\bar{C}$ defined here, and (4.21) reduces to :

$$H \left[\bar{U}_L + \bar{V}_R + \bar{C}_L - \bar{C}_R \right] \neq 0. \qquad (4.24)$$

If e is even, substitute $\bar{C}$ from (4.26a) into (4.24). Regarding the resulting n - tuple as a burst, as shown below * ,

$$\left[u_1, u_2, .., u_s, \overbrace{1\,0\,..\,0}^{S+1}, .., .., \overbrace{1\,0\,..\,0}^{S+1}, 0\,..\,0, v_{n-r+1} .., v_{n-1}, v_n \right] \quad s, r, \leq S$$

it is seen that (4.24) is satisfied if

$$1 + \frac{e}{2}(S+1) + (s+r) \leqslant n - k \cdot\cdot$$

which gives

$$S \leqslant \frac{n - k - e/2 - 1}{e/2 + 2} \quad , \quad e \text{ even} \quad . \qquad (4.33)$$

The remainder of the proof can now be obtained by comparing with the original proof.

Q.E.D.

---

* For clarity, $\bar{C}_R$ is not shown, since it is overlapped by the other three components $\bar{U}_L$, $\bar{V}_R$ and $\bar{C}_L$.

In this chapter, a selected number of theorems on the detection and correction of slip have been re-examined using the vector - matrix representation. For a more extensive treatment, the reader is referred to the paper by Tavares and Fukada.[55] There, the theorems are proved without relying explicitly on the fact that an (n , k) cyclic code word can not be a burst of length (n - k) or less. However, this omission tends to result in longer proofs and requires the introduction of a couple lemmas.

Stiffler[52] also proves Theorems 2.1, 2.2 and Corollary 2.1 using the vector - matrix representation. However, the proofs given here are somewhat different from Stiffler's. His first result is on the detection of slip for group codes in general, and he then considers cyclic codes as special cases of group codes. The result for group codes in general is somewhat awkward to apply, and worthwhile simplifications result by taking advantage of the properties of cyclic codes from the outset, as is done here.

CHAPTER V

LOSS OF SYNCHRONIZATION IN THE PRESENCE OF BURST ERRORS FOR COSET CODES

5.1 Introduction

In the previous chapters, the additive errors were assumed to occur independently. This assumption is often made in the literature on error correcting codes.[39] An important class of cyclic codes, the B C H codes[6,7] are known to be quite effective in combatting this type of error.[39]

Another important class of errors are burst errors.[16,39] The concept of a burst has been used in earlier chapters but for reference, a definition of a burst is now given :

Definition : A burst of length T is an n - tuple which is zero everywhere except in an interval of length T, the first and last positions of which are non-zero.

Experimentally, it has been found that the type of errors occurring on telephone lines may be reasonably represented by burst errors or even better, by a combination of burst errors and independent errors.[1,17] In general, a channel which suffers from impulse noise or fading (of signal strength) will tend to have burst errors. Further justification for the study of burst errors will not be given, as their study is now well established in the literature.[1,16,17,39] In fact, special codes have been developed with the explicit purpose of detecting and correcting burst errors. A well known class of such codes are the Fire Codes.[16]

In this chapter, the ability of cyclic codes to combat slip in the presence of burst errors will be studied. In addition, their ability to handle slip and multiple - adjacent - additive - errors will also be considered. Such errors are a subset of the class of burst errors. In the next chapter, detection and correction of slip in the presence of burst errors will be considered for Fire Codes. Their performance can then be compared with cyclic codes in general.

5.2 Slip In The Presence Of Multiple - Adjacent - Additive - Errors

For reference, a multiple - adjacent - additive - error of length T will be defined as an n - tuple which has T consecutive non-zero terms and is zero elsewhere. It is seen from the definition that it is a burst of length T which has weight T.

For an (n, k) cyclic code, k or more consecutive zeros can only occur in the zero word. This follows from the fact that a non-zero word cannot be a burst of length $n - k$ or less. It can also be shown that $k + 1$ or more consecutive ones can only occur in an (n, k) cyclic code if the code contains the word of weight n.

A burst of T adjacent - additive - errors which begins at the $(p + 1)$'th position of the n - tuple may be described by the polynomial $x^p (1 + x + \ldots + x^{T-1})$. Let d be the minimum distance of the (n, k) cyclic codes considered in this chapter. The next theorem is valid only for binary cyclic codes.

Theorem 5.1

Any (n, k) binary cyclic code has a coset code which can detect the simultaneous occurrence of T or less adjacent - additive - errors and S or less bits of slip, where

$$(1) \qquad S = \text{Min.} \left\{ d - 6 \, , \; n - T - 3 \right\}$$

or

$$(2) \qquad S + T = \left[3 \, (d - 3) / 2 \right]$$

but the decoder cannot determine the nature of the error.

Proof : Let $C(x) = 1 + x^{n-1}$. Part (1). Assume that s bits of left slip, $s \leq S$, occurs to the coset code word $W(x) + C(x)$. Also, let $x^p (1 + x + \ldots x^{t-1})$ represent a single burst of t adjacent - additive - errors, $t \leq T$, where the burst begins at the $(p + 1)$'th coordinate. Then, the decoder can detect the presence of error if and only if *

$$\left\{ x^s (W(x) + C(x)) + C(x) + x^p (1 + x + \ldots + x^{t-1}) + U_s(x) \right\} \neq 0 \qquad (5.1)$$

for all $s \leq S$, $t \leq T$, $p \leq n$. Since $\left\{ x^s W(x) \right\} = 0$, (5.1) becomes

$$\left\{ C(x) (x^s + 1) + x^p (1 + x + \ldots + x^{t-1}) + U_s(x) \right\} \neq 0. \qquad (5.2)$$

Let $Q(x)$ be the polynomial in (5.2), then $Q(x)$ is a cyclic code word only if $x\,Q(x)$ and $x\,Q(x) + Q(x)$ are also cyclic code words. Now,

$$x\,Q(x) = x\,C(x) (x^s + 1) + x^p (x + x^2 + \ldots x^t) + x\,U_s(x) \qquad (5.3)$$

and therefore

---

* Note that for binary codes, addition and subtraction are identical.

$$Q(x)(x+1) = C(x)(x^s+1)(x+1) + x^p(1+x^t) + U_s(x)(x+1) \tag{5.4}$$

and substituting $C(x) = 1 + x^{n-1}$ in (5.4) gives

$$Q(x)(x+1) = x^{n-1} + x^{s+1} + x^p(1+x^t) + U_{s+1}(x) \tag{5.5}$$

where $U_{s+1}(x)$ "absorbs" all terms of degree s or less. Now, (5.2) is satisfied if $\{Q(x)(x+1)\} \neq 0$, i.e.,

$$\{x^{n-1} + x^p(1+x^t) + x^{s+1} + U_{s+1}(x)\} \neq 0 . \tag{5.6}$$

The maximum weight of (5.6) is $4+(S+1)$. Therefore, if

$$d > 4 + S + 1$$

or

$$S \leq d - 6$$

(5.6) will be satisfied. In addition, the polynomial in (5.6) cannot be the zero word if $n > S + T + 2$, which implies $S + T \leq n - 3$.

Part (2). The maximum weight of (5.2) is $T + S + 2$, since $C(x)(x^s+1) = x^{n-1} + x^s + x^{s-1} + 1$, and $U_s(x)$ "absorbs" $x^{s-1} + 1$. With this absorption in mind, (5.2) becomes

$$\{x^{n-1} + x^p(1 + x + \ldots + x^{t-1}) + x^s + U_s(x)\} \neq 0 . \tag{5.7}$$

Now, suppose m terms are missing in $U_s(x)$, then the maximum weight of the polynomial in (5.7) is $2 + t + (s - m)$. Therefore if

$$2 + t + (s - m) < d$$

i.e.

$$2 + t + s < d + m \qquad (5.8)$$

(5.7) is satisfied. Also, if m terms of $U_s(x)$ are missing, the maximum weight of (5.6) is $4 + 2m$, therefore (5.6) is satisfied if

$$2m + 4 < d . \qquad (5.9)$$

If either (5.8) or (5.9) is satisfied, the decoder can detect an error. Eliminating m from (5.8) and (5.9) gives

$$s + t < \frac{3d}{2} - 4 . \qquad (5.10)$$

When s and t assume their maximum values, (5.10) becomes

$$S + T = \left[3(d-3)/2\right] .$$

By the lemma in Appendix II, the condition $S + T \leq n - 3$ is redundant for condition (2) since

$$3(d-3)/2 \leq n - 3 .$$

A similar proof applies for right slip.

Q.E.D.

Theorem 5.2

Any (n , k) binary cyclic code has a coset code which can correct T or less adjacent - additive - errors and S or less bits of slip, if both errors do not occur simultaneously, where either

$$(1) \qquad S = \text{Min.} \left\{ \left[(3\ (d-3)/2) - T \right] ,\ \left[(n-k-2)/2 \right] \right\} ,$$

or

$$(2) \qquad S = \text{Min.} \left\{ d-6,\ n-T-3,\ \left[(n-k-2)/2 \right] \right\} .$$

Proof : Let $C(x) = 1 + x^{n-1}$. First, it will be shown that the syndromes for slip are different from the syndromes for a burst of t adjacent - additive - errors, $(t \leq T)$. Assuming s bits of left slip, it is sufficient to show that

$$\left\{ x^s (W_1(x) + C(x)) + C(x) + U_s(x) \right\} \neq \left\{ W_2(x) + x^p (1 + x + \ldots + x^{t-1}) \right\} \quad (5.11)$$

where $W_1(x)$ and $W_2(x)$ are any two cyclic code words, and $s \leq S$, $t \leq T$. Putting all terms on the left, (5.11) simplifies to

$$\left\{ C(x)\ (x^s + 1) + x^p (1 + x + \ldots + x^{t-1}) + U_s(x) \right\} \neq 0 . \qquad (5.12)$$

A comparison of (5.12) and (5.2) will show that they are identical and hence (5.12) is satisfied if

$$(1) \qquad S = \text{Min.} \left\{ d-6,\ n-T-3 \right\}$$

or

$$(2) \qquad S + T = \left[3\ (d-3)/2 \right] .$$

Next, it is shown that the syndromes for left and right slip are distinct, i.e.,

$$\left\{ x^s (W_1(x) + C(x)) + C(x) + U_s(x) \right\} \neq \left\{ x^{-r} (W_2(x) + C(x)) + C(x) + U_{-r}(x) \right\} \quad (5.13)$$

for all $s, r \leq S$, where $W_1(x)$ and $W_2(x)$ are arbitrary cyclic code words. Now,

(5.13) simplifies as

$$\left\{C(x)\,(x^{s}+x^{-r}) + U_{s}(x) + U_{-r}(x)\right\} \neq 0 \qquad (5.14)$$

and (5.14) is satisfied if and only if

$$\left\{C(x)\,(x^{s+r}+1) + U_{s+r}(x)\right\} \neq 0 \qquad (5.15)$$

which is obtained by cyclically shifting (5.14) by r bits (multiply through by x^{r}), and setting $U_{s+r}(x) = x^{r} U_{s}(x) + U_{r}(x)$. Substituting $C(x) = 1 + x^{n-1}$ in (5.15) reveals that it is identical to (2.23), and hence (5.15) is true if $S \leq (n-k-2)/2$.

The decoder can correct any burst of t adjacent - additive - errors (t T) if and only if

$$\left\{E_{1}(x)\right\} \neq \left\{E_{2}(x)\right\} \qquad (5.16)$$

for all $W\left[E_{i}(x)\right] \leq T$, $i = 1, 2$, where $E_{1}(x)$ and $E_{2}(x)$ are two such bursts. Putting all terms on the left, (5.16) becomes

$$\left\{E_{1}(x) + E_{2}(x)\right\} \neq 0 \qquad (5.17)$$

and it is not hard to show that (5.17) is true if $d > 4$. But $d > 4$ is implied for non-trivial values of S (i.e. $S \neq 0$) in the theorem.

Q.E.D.

5.3 Detection Of Slip In The Presence Of Burst Errors

In this section, it is assumed that additive errors occur in bursts, where the term "burst" is used in the general sense as defined at the beginning of this chapter. In the next theorem[57], slip and additive error are allowed to occur simultaneously.

Theorem 5.3

Any (n, k) binary cyclic code has a coset code which can detect the simultaneous occurrence of a single burst of length T or less and S or less bits of slip, where $T + S = [3(d-3)/2]$.

Proof : Let $C(x) = 1 + x^{n-1}$, and $E_t(x) = x^p(1+e_1x + \ldots + e_{t-2}x^{t-2}+x^{t-1}), e_i=0,1,$ be a burst of length t which begins in the $(p+1)$ th position, $p < n$. Assuming left slip of s bits, it is sufficient to show that

$$\left\{x^s (W(x) + C(x)) + C(x) + U_s(x) + E_t(x)\right\} \neq 0 \qquad (5.18)$$

where $s \leq S$, and $W(x)$ is any cyclic code word. Now, (5.18) simplifies to

$$\left\{C(x)(x^s + 1) + U_s(x) + E_t(x)\right\} \neq 0 \qquad (5.19)$$

and substituting for $C(x)$ in (5.19) gives

$$\left\{x^{n-1} + x^s + U_s(x) + E_t(x)\right\} \neq 0 \qquad (5.20)$$

where $U_s(x)$ absorbs terms of degree less than s. The polynomial $x^{n-1} + x^s + U_s(x)$ may

be regarded as a burst of length $s + 2$. Letting $P_{s+2}(x) = x^{n-1} + x^s + U_s(x)$, (5.20) becomes

$$\left\{P_{s+2}(x) + E_t(x)\right\} \neq 0 \tag{5.21}$$

and so the problem reduces to showing that the sum of two bursts is not a cyclic code word. Let there be a total of m terms missing in the two bursts $P_{s+2}(x)$ and $E_t(x)$ and define $Q(x) = P_{s+2}(x) + E_t(x)$. Then,

$$W\left[Q(x)\right] \leq s + 2 + t - m$$

and (5.21) is satisfied if

$$s + t + 2 - m < d \quad . \tag{5.22}$$

Also, if

$$\left\{x\,Q(x) + Q(x)\right\} \neq 0 \tag{5.23}$$

then (5.21) is true. But the maximum weight of $Q(x)(x + 1)$ is $2m + 4$ (this fact can be established by checking the various possibilities), hence (5.23) is satisfied if

$$2m + 4 < d \tag{5.24}$$

If either (5.22) or (5.24) is true, the decoder will detect an error. Eliminating m from (5.22) and (5.24) gives

$$t + s < \frac{3d}{2} - 4 \tag{5.25}$$

and taking maximum values of t and s gives

$$T + S = \left[3 (d - 3) / 2\right] .$$

A similar proof applies for right slip .

Q.E.D.

During the proof of Theorem 5.3, a useful intermediate result was also proved. It concerns the ability of a binary cyclic code to detect a pair of burst errors, or equivalently to correct a single burst error. Changing (5.21) slightly, it is required to prove that

$$\left\{E_1 (x) + E_2 (x)\right\} \neq 0 \qquad (5.26)$$

where $E_1 (x)$ and $E_2 (x)$ are bursts of length t_1 and t_2, respextively. The solution of (5.21) was given in (5.25) and the corresponding solution for (5.26) is

$$t_1 + t_2 < \frac{3 d}{2} - 2 . \qquad (5.27)$$

Note that t_1 replaces $s + 2$. Letting T_1 and T_2 be the maximums of t_1 and t_2, respectively, (5.27) becomes

$$T_1 + T_2 = \left[(3 d - 5) / 2\right] . \qquad (5.28)$$

To correct a single burst, it is necessary that $T_1 = T_2 = T$ and (5.28) becomes

$$T = \left[(3 d - 5) / 4\right] .$$

The above results are summarized in two lemmas.

Lemma 5.1

Any (n , k) binary cyclic code, with minimum distance d, can detect any two bursts of length t_1 and t_2 if $t_1 + t_2 \leq \left[(3 d - 5) / 2\right]$.

Lemma 5.2

Any (n , k) binary cyclic code, with minimum distance d, can correct all single bursts of length T or less, where $T = \left[(3d - 5) / 4\right]$.

5.4 Correction Of Slip In The Presence Of Burst Errors

In the next two theorems[57] it will be assumed that slip and additive error (burst error) do not occur simultaneously. For these theorems, the decoder can distinguish between slip and additive error. In the first result, the decoder corrects the slip but only detects the additive error, but in the second result it corrects both types of error.

Theorem 5.4

Any (n , k) binary cyclic code has a coset code which can correct S or less bits of slip and detect a burst of length T or less, if slip and additive error do not occur simultaneously, where

$$S = \text{Min.} \left\{\left[3 (d - 3) / 2\right] - T \quad , \quad \left[(n - k - 2) / 2\right]\right\}.$$

Proof : Let $C (x) = 1 + x^{n-1}$. To correct slip, it is necessary to show that the syndromes for left and right slip are distinct. In addition, the decoder must distinguish between slip and additive error. The latter statement is proved first.

Assuming s bits of left slip , it is sufficient to prove that

$$\left\{x^s (W_1 (x) + C (x)) + C (x) + U_s (x)\right\} \neq \left\{W_2 (x) + E_t (x)\right\} \qquad (5.29)$$

for all $s \leq S$, $t \leq T$, where $W_1(x)$ and $W_2(x)$ are any two cyclic code words and $E_t(x)$ is a burst of length t . Putting all terms on the left, (5.29) simplifies to

$$\left\{ C(x)(x^s + 1) + U_s(x) + E_t(x) \right\} \neq 0 \quad . \tag{5.30}$$

Comparing (5.30) with (5.19) reveals that they are identical and hence (5.30) is satisfied if (5.25) is true, which gives

$$S \leq \left[3(d - 3)/2 \right] - T \quad . \tag{5.31}$$

The above result is also true for right slip. It remains to show that the syndromes for left and right slip are distinct, or equivalently, that

$$\left\{ x^s (W_1(x) + C(x)) + U_s(x) \right\} \neq \left\{ x^{-r} (W_2(x) + C(x)) + U_{-r}(x) \right\} \tag{5.32}$$

for all $r, s \leq S$, where r and s are the magnitudes of the right and left slip, respectively. It is easy to show that (5.32) reduces to (2.23) after putting $C(x) = 1 + x^{n-1}$. It follows that (2.23) , and hence (5.32) is satisfied if

$$S \leq \left[(n - k - 2)/2 \right] \quad . \tag{5.33}$$

Also, as shown for Theorem 2.4, the decoder can determine the magnitude as well as the direction of the slip. S is now chosen as the minimum of (5.31) and (5.33).

Theorem 5.5

Any (n , k) binary cyclic code has a coset code which can correct S or less bits of slip and a single burst of length T or less, if slip and additive error do not

occur simultaneously, where,

$$(1) \qquad T \leq \left[3\,(d-5)/4\right]$$

and

$$(2) \qquad S = \text{Min.}\left\{\left[3\,(d-3)/2\right] - T \;,\; \left[(n-k-2)/2\right]\right\}.$$

Proof : Let $C(x) = 1 + x^{n-1}$, and recall the proof of Theorem 5.4. All that remains to be proved is the condition under which single burst errors (no slip present) have distinct syndromes. However, an answer to this is given by Lemma 5.2, i.e. to correct all single burst errors of length T, it is necessary that $T \leq \left[(3d-5)/4\right]$.

Q.E.D.

In the next theorem[57], slip and burst errors are allowed to occur simultaneously.

Theorem 5.6

Any (n , k) binary cyclic code has a coset code which can correct the simultaneous occurrence of S or less bits of slip and a single burst of length T or less, where

$$S + T = \left[3\,(d-7)/4\right].$$

Proof : Let $C(x) = 1 + x^{n-T-3} + x^{n-2} + x^{n-1}$ * It is sufficient to show that the syndromes for left slip plus a burst error are different from the syndromes for right slip plus a burst error, i.e., that

---

* By symmetry, it is easily seen that another suitable choice is $C(x) = 1 + x + x^{T+2} + x^{n-1}$.

$$\left\{ x^{s} \left(W_1 (x) + C (x)\right) + U_s (x) + E_{t1} (x) \right\}$$

$$\neq \left\{ x^{-r} \left(W_2 (x) + C (x)\right) + U_{-r} (x) + E_{t2} (x) \right\} \qquad (5.34)$$

for all $s , r \leq S ; \; t_1 , t_2 \leq T$, where the cyclic code word $W_1 (x)$ suffers s bits of left slip and a burst of length t_1 and $W_2 (x)$ suffers r bits of right slip and a burst of length t_2. Putting all terms of (5.34) on the left and simplifying, gives

$$\left\{ C (x) \left(x^{s} + x^{-r}\right) + U_s (x) + U_{-r} (x) + E_{t1} (x) + E_{t2} (x) \right\} \neq 0 . \qquad (5.35)$$

Performing a cyclic shift of r bits, (5.35) is true if and only if

$$\left\{ C (x) \left(x^{s+r} + 1\right) + U_{s+r} (x) + E_{t1} (x) + E_{t2} (x) \right\} \neq 0 \qquad (5.36)$$

where $U_{s+r} (x) = x^{r} U_s (x) + U_r (x)$ and the cyclic shift has only a trivial effect on $E_{t1} (x)$ and $E_{t2} (x)$ and so is ignored. It must be verified that the $C (x)$ chosen will satisfy the requirements of the theorem. It is sufficient to show that the polynomial in (5.36) is not a cyclic code word, including the zero word. Written as an n - tuple, $C (x)$ has the form

$$\underbrace{[1, 0 \ldots\ldots\ldots, 0 , 1 , 0 \ldots\ldots, 0 , 1 , 1]}_{T+3} \qquad (5.37)$$

In this proof the algebra looks complicated, but the argument is sometimes clearer if the n - tuples or "pictures" are kept in mind. Now ,

$$C (x)(x^{s+r} + 1) = 1 + x^{n-T-3} + x^{n-2} + x^{n-1} + x^{s+r} \left(1 + x^{n-T-3} + x^{n-2} + x^{n-1}\right) \qquad (5.38)$$

Since $U_{s+r} (x)$ is a random polynomial of degree less than $s + r$, it can absorb all terms of degree less than $s + r$, and in an expression involving $U_{s+r} (x)$, such terms will simply

be omitted. Hence, write

$$C(x)(x^{s+r}+1)+U_{s+r}(x)=x^{n-1}+x^{n-2}+x^{n-T-3}+x^{s+r-(T+3)}+x^{s+r}+U_{s+r}(x) \tag{5.39}$$

and (5.39) has the form (omitting $x^{s+r-(T+3)}$)

$$[u_o, u_1, \ldots\ldots, u_{s+r}, 1, 0, \ldots\ldots 0, \overbrace{1, 0, \ldots\ldots 0, 1, 1}^{T+3}]. \tag{5.40}$$

The term $x^{s+r-(T+3)}$ is omitted because it may occupy one of several places. For instance, if $s+r \geq T+3$, it will be absorbed by $U_{s+r}(x)$, and if $s+r-(T+3)=-1$, it cancels x^{n-1} and so on. Substituting (5.39) into (5.36), it can be shown that for all $E_{t1}(x)$ and $E_{t2}(x)$, where $t_1, t_2 \leq T$, the polynomial in (5.36), cannot become the zero word. Calling the polynomial in (5.36) $Q(x)$, it remains to be shown that $Q(x)$ cannot be a non-zero word.

The maximum weight of $Q(x)$ is

$$(s+r+5)+T+T.$$

Let

$$s+r+5+2T=d+m \tag{5.41}$$

hence when more than m terms are missing from $U_{s+r}(x)+E_{t1}(x)+E_{t2}(x)$, $Q(x)$ has weight less than d and cannot be a code word. Assuming m terms are missing, the maximum weight of $xQ(x)+Q(x)$ is found to be $2m+10$, and if

$$2m+10<d \tag{5.42}$$

then (5.46) is satisfied (since if $Q(x)(x+1)$ is not a cyclic code word, then $Q(x)$ is

not a cyclic code word). If there are less than m terms missing in Q (x), then from (5.42), (5.36) is satisfied. However, if more than m terms are missing, (5.36) is again satisfied, based on (5.41). Eliminating m from (5.41) and (5.42) gives

$$s + r + 2T + 5 < d + \left(\frac{d}{2} - 5\right)$$

or

$$S + T \leq [3(d-7)/4] \qquad (5.43)$$

where S is the maximum value of both s and r . Finally, a necessary condition such that the polynomial in (5.36) cannot become the zero word is that $E_{t1}(x) + E_{t2}(x)$ cannot cancel all the non-zero terms in (5.40). The condition is

$$n \geq 2S + 2T + 3$$

or

$$2S + 2T \leq n - 3 \; . \qquad (5.44)$$

It is not hard to show that (5.43) implies (5.44) .

Q.E.D.

The next theorem describes a range of slip in which no coset code can correct the simultaneous occurrence of slip and a single burst error. This result is not restricted to binary codes.

Theorem 5.7

No coset of any (n , k) cyclic code can simultaneously correct a single burst of length T and s bits of slip, when

$$(n - k - 2T) / 2 \leq s \leq k + T \ .$$

Proof : It is sufficient to show that the inequality (5.36) cannot be guaranteed for all $U_{s+r}(x)$, $E_{t1}(x)$ and $E_{t2}(x)$ when slip s, lies in the range given in the theorem. Rewrite (5.36) as

$$\left\{C(x)(x^{s+r} - 1)\right\} \neq \left\{U_{s+r}(x) + E_t(x)\right\} \qquad (5.45)$$

where $E_t(x) = E_{t1}(x) + E_{t2}(x)$, $t \leq 2T$, and the negative sign on the left allows for the fact that the code may be non-binary. Comparing (5.45) with (3.57) it is seen that they are the same except that here the additive errors are burst errors. However, in spite of this difference, the same arguments are valid and the results are identical. Hence the result is obtained by replacing e by T in Theorem 3.7.

Q.E.D.

In Table V-1, some numerical results are presented for the theorems of this chapter, using as examples the same cyclic codes as was used in Table III - 1.

To the best of the author's knowledge, all the theorems presented in this chapter are new.

TABLE V - 1

A Short List Of Cyclic Codes To Illustrate Their Ability To Detect Or Correct Slip, In The Presence Of A Burst Of Additive Errors Of Length T Or Less.

CYCLIC CODE (n , k)	d	T	MAXIMUM SLIP (S) FOR THEOREMS LISTED						
			5.1	5.2	5.3	5.4	5.5	5.6	5.7*
(23 , 12)	7	1	5	4	5	4	4	0	4
		2	4	4	4	4	4	0	3
		3	3	3	3	3	3	0	2
(127 , 85)	13	1	14	14	14	14	14	3	19
		2	13	13	13	13	13	2	18
		3	12	12	12	12	12	1	17
		11	7	7	4	4	0	0	9
		12	7	7	3	3	0	0	8
(255 , 191)	17	1	20	20	20	20	20	6	30
		2	19	19	19	19	19	5	29
		3	18	18	18	18	18	4	28
		4	17	17	17	17	17	3	27
		16	11	11	5	5	0	0	15
		18	11	11	3	3	0	0	13
(255 , 163)	25	1	32	32	32	32	32	12	44
		2	31	31	31	31	31	11	43
		3	30	30	30	30	30	10	42
		4	29	29	29	29	29	9	41
		21	19	19	12	12	0	0	24
		22	19	19	11	11	0	0	23
		23	19	19	10	10	0	0	22
(127 , 22)	47	1	65	51	65	51	51	29	-
		2	64	51	64	51	51	28	-
		3	63	51	63	51	51	27	-
		4	62	51	62	51	51	26	-
		15	51	51	51	51	51	15	-
		16	50	50	50	50	50	14	-
		32	41	41	34	34	34	0	-
		33	41	41	33	33	33	0	-
		40	41	41	26	26	0	0	-

* Obtained by setting $S = [(n - k - 2T - 1)/2]$. This S serves as an upper bound on the performance of Theorem 5. 6.

CHAPTER VI

LOSS OF SYNCHRONIZATION FOR COSET CODES OF FIRE CODES

6.1 Introduction

Fire codes[16, 39] are a class of cyclic codes which are quite effective in detecting and correcting burst errors. However, they have small minimum distance and are not suitable for handling independent additive errors.

In this chapter, it will be assumed that the additive errors are burst errors and advantage will be taken of the special abilities of the Fire Codes. Recall that one consequence of a slip of s bits is the introduction into the receiver frame of a random polynomial of length s which can be considered as a burst of length s or less. It may thus be suspected that Fire Codes have an advantage in handling slip errors.

6.2 A Brief Review of Fire Codes

An (n,k) Fire code may be defined as a cyclic code with generator polynomial G(x) given by[39]

$$G(x) = P(x)(x^a - 1) \tag{6.1}$$

where P(x) is an irreducible polynomial of degree b which has exponent* β, and a is not divisible by β. The length n, of the code, is the least common multiple of a and β. From the above definitions, it follows that the number of check bits is a+b, i.e., $n-k = a+b$.

---

* An irreducible polynomial has exponent β if it divides $(x^\beta - 1)$ but not $(x^p - 1)$ for all $p < \beta$.

The expression "A Fire code with parameters (a,b)" will be used, where a and b are defined above. In the results to follow, the properties of Fire codes stated in Chapter 10 of Peterson[39] will be assumed. In particular, Theorem 10.1 in Peterson[39] is quoted, without proof, for easy reference.

Theorem 6.1

A vector that is the sum of a burst of length T_1 or less and a burst of length T_2 or less, where $T_2 \geq T_1$, cannot be a code word in a Fire code having parameters (a,b) if

$$T_1 + T_2 \leq a+1 \text{ and } T_1 \leq b. \tag{6.2}$$

Two corollaries follow from Theorem 6.1[39],

Corollary 6.1

A Fire code can correct a single burst error of length T_1 or less and in addition detect any burst of length $T_2 > T_1$, if

$$T_1 + T_2 \leq a+1 \text{ and } T_1 \leq b .$$

Corollary 6.2

A Fire code can detect any two bursts having lengths T_1 or less and T_2 or less, where $T_2 \geq T_1$, if

$$T_1 + T_2 \leq a+1 \text{ and } T_1 \leq b.$$

In addition, since any (n,k) cyclic code can detect a single burst of length $n - k$ or less, a Fire code with parameters (a,b) can detect any single burst error of length

$a + b$ or less, since $n - k = a + b$.

6.3 Detection of Slip for Coset Codes of Fire Codes

The first result of this section concerns the ability of coset codes of Fire codes to detect the simultaneous occurrence of slip and a single burst error.

Theorem 6.2

Any Fire code, with parameters (a,b) has a coset code which can detect the simultaneous occurrence of S or less bits of slip and a single burst of length T or less if $S + T \leq a - 1$ and either $S \leq b - 2$ or $T \leq b$, but the decoder cannot determine the nature of the error.

Proof

Let $C(x) = 1 + x^{n-1}$. Also, let $W(x)$ be any code word of the Fire code and assume the simultaneous occurrence of s bits of left slip and a burst error $E_t(x)$ of length t. It is sufficient to show that

$$\left\{x^s (W(x) + C(x)) + U_s(x) + E_t(x) - C(x)\right\} \neq 0 \qquad (6.3)$$

for all $s \leq S$, $t \leq T$. Noting that $\left\{x^s W(x)\right\} = 0$, (6.3) becomes

$$\left\{C(x)(x^s - 1) + U_s(x) + E_t(x)\right\} \neq 0 . \qquad (6.4)$$

Putting $C(x) = 1 + x^{n-1}$ in (6.4) and recalling that $U_s(x)$ absorbs all terms of degree less than s, gives

$$\left\{U_s(x) + x^s + x^p\,(1 + e_1 x + \ldots + e_{t-2}x^{t-2} + x^{t-1}) - x^{n-1}\right\} \neq 0, \quad e_i \in F. \tag{6.5}$$

Written as an n-tuple, the polynomial in (6.5) has the form

$$(u_o, u_1, \ldots, u_{s-1}, 1, 0, \ldots, 0, 1, e_1, e_2, \ldots, e_{t-2}, 1, 0, \ldots, -1). \tag{6.6}$$

We can regard (6.6) as the sum of two bursts, $E_t(x)$ of length t and $(U_s(x) + x^s - x^{n-1})$ of length $s + 2$. Hence by Corollary 6.2, if $(s + 2) + t \leq a+1$ and either $s + 2 \leq b$ or $t \leq b$, then (6.3) will be satisfied. It is also necessary that $n \geq s+t+2$, but this is true if $s + t + 2 \leq a+1$. A similar proof applies to right slip.

Q.E.D.

To obtain an upper bound on the ability of coset codes of Fire codes to detect the simultaneous occurrence of slip and a single burst, write (6.4) as

$$\left\{C(x)\,(x^s - 1)\right\} \neq \left\{U_s(x) + E_t(x)\right\}. \tag{6.7}$$

For a Fire code, $n - k = a + b$, thus when $s+t \geq a+b$, $U_s(x) + E_t(x)$ can generate all possible syndromes. Hence, regardless of the choice of C(x), (6.7) cannot be guaranteed when $s+t \geq a+b$. This result is stated in the next theorem.

Theorem 6.3

No coset of any Fire code with parameters (a,b) can detect (with certainty) the simultaneous occurrence of S bits of slip and a burst of length T, if $S + T \geq a+b$.

Theorem 6.3 is true for all choices of C(x). It is interesting to consider the upper bound for the particular choice of $C(x) = 1 + x^{n-1}$ in Theorem 6.2. Such a bound can be found by considering the n-tuple (6.6). A cyclic shift of one bit on (6.6) gives

$$(-1, u_0, u_1, \ldots, u_{s-1}, 1, 0, \ldots, 0, 1, e_1, e_2, \ldots, e_{t-2}, 1, 0, \ldots, 0) \tag{6.8}$$

which can be regarded as the sum of a burst of length s+2 and a burst of length t. Now, the inequality (6.5) fails when (6.6) is a Fire code word. The generator polynomial $G(x) = P(x)(x^a - 1)$ is itself a Fire code word and can be regarded as the sum of two bursts, each of length b+1, i.e., $x^a P(x)$ and $P(x)$ which may or may not overlap. It is clear from (6.8) that when $S + 2 \geq b+1$ and $T \geq b+1$, (6.6) can become G(x). Hence, the coset given by $C(x) = 1 + x^{n-1}$ will be vulnerable to the simultaneous occurrence of S bits of slip and a burst of length T, when $S \geq b-1$ and $T \geq b+1$. Now these two inequalities imply that $S + T \geq 2b$ (but not vice versa). Compare this result with Theorems 6.2 and 6.3.

In some situations, it may be desirable to correct a single burst error and merely detect the presence of slip, assuming that they do not occur simultaneously. The next result applies to this case.

Theorem 6.4

Any Fire code, with parameters (a,b), has a coset code which can correct a burst of length T or less and also detect S or less bits of slip, if burst errors and slip do not occur simultaneously, where

$$2 < T \leq b \quad \text{and} \quad S \leq a - 2T + 1.$$

Proof

Let $C(x) = 1 + x^{T-2} + x^{n-1}$*. It is necessary to distinguish between a burst

* When T = 1,2, choose $C(x) = 1 + x^{n-1}$.

error of length T or less and a slip of S or less bits. This can be achived if a slip of s bits always generates a burst of length T + 1 or greater, for all $s \leq S$. It is sufficient to show that

$$\left\{ x^s(W_1(x) + C(x)) + U_s(x) - C(x) \right\} \neq \left\{ W_2(x) + E_t(x) \right\} \qquad (6.9)$$

where the word $W_1(x)$ suffers s bits of left slip and $W_2(x)$ has a burst error $E_t(x)$ where t T. Making the usual simplifications, (6.9) can be written as

$$\left\{ C(x)(x^s - 1) + U_s(x) \right\} \neq \left\{ E_t(x) \right\}. \qquad (6.10)$$

Substituting $C(x) = 1 + x^{T-2} + x^{n-1}$ in the expression on the left of (6.10) gives

$$U_s(x) + x^s - x^{T-2} + x^{s+T-2} - x^{n-1} \quad .$$

The above polynomial can be viewed as a burst of length s + T and thus a left slip of s bits generates a burst of length T+1 or greater. Putting $E_t(x)$ on the left in (6.10) gives

$$\left\{ C(x)(x^s - 1) + U_s(x) + E_t(x) \right\} \neq 0 \quad . \qquad (6.11)$$

The polynomial in (6.11) is the sum of two bursts of total length (s+T) + t. Hence, by Corollary 6.1, if $T \leq b$ and $S + 2T \leq a+1$, the decoder can correct any burst of length T or less and detect S or less bits of slip, where $S+2T \leq a+1$.

For right slip of r bits, (6.9) becomes

$$\left\{ C(x)(x^{-r} - 1) + U_{-r}(x) \right\} \neq \left\{ E_t(x) \right\} .$$

This is equivalent to

$$\left\{ C(x)(1 - x^r) + U_r(x) + E_t(x) \right\} \neq 0 \qquad (6.12)$$

which has the same form as (6.11). Hence the same result applies to right slip.

The choice of C(x) in this theorem implies that $T > 2$. This is not a real restriction since a burst of length two (i.e. a double-adjacent-error) or one is trivial. Also, multiple-adjacent-errors were examined in Chapter V.

Q.E.D.

In Theorem 6.4, an acceptable alternative choice for C(x) is $C(x) = 1 + x^{n-T+1} + x^{n-1}$. These choices of C(x) are optimum in the sense that a slip of one bit generates a burst of length T + 1, which is the minimum required.

6.4 Correction of Slip for Coset Codes of Fire Codes

In the next theorem, both slip and a burst error are corrected, but it is still assumed that they cannot occur simultaneously.

Theorem 6.5

Any Fire code, with parameters (a,b), has a coset code which can correct a burst of length T or less and S or less bits of slip, if burst errors and slip do not occur simultaneously, where

$$2 < T \leq b \text{ and } S = \text{Min.} \left\{ a - 2T + 1, \left[(a+b-T)/2 \right] \right\} .$$

Proof

Let $C(x) = 1 + x^{T-2} + x^{n-1}$ as in Theorem 6.4, and hence assume the results proved there. All that remains to be done is to determine the conditions under which the decoder can distinguish between right and left slip, i.e.,

$$\left\{x^s (W_1(x) + C(x)) + U_s(x)\right\} \neq \left\{x^{-r} (W_2(x) + C(x)) + U_{-r}(x)\right\} \qquad (6.13)$$

where the terms are defined as usual. The above reduces to

$$\left\{C(x)(x^s - x^{-r}) + U_s(x) + U_{-r}(x)\right\} \neq 0 \quad . \qquad (6.14)$$

Performing a cyclic shift of r bits on (6.14) gives the equivalent relation

$$\left\{C(x)(x^{s+r} - 1) + U_{s+r}(x)\right\} \neq 0 \quad . \qquad (6.15)$$

Substituting $C(x) = 1 + x^{T-2} + x^{n-1}$ in (6.15) and noting that $U_{s+r}(x)$ absorbs all terms of degree less than s+r gives

$$\left\{U_{s+r}(x) + x^{s+r} - x^{T-2} + x^{s+r+T-2} - x^{n-1}\right\} \neq 0 \quad . \qquad (6.16)$$

The polynomial in (6.16) is a burst of length s+r+T, hence (6.16) is satisfied if $2S + T \leq n-k = a+b$

In all, the conditions to be satisfied are*

(1) $T \leq b$,

(2) $S + 2T \leq a+1$ and

(3) $2S + T \leq a+b$.

They may be rearranged to give

(1) $T \leq b$

and

(2) $S = \text{Min.}\left\{a - 2T + 1, \left[(a+b-2T)/2\right]\right\}$.

Q.E.D.

---

* When T = 1 or 2, choose $C(x) = 1 + x^{n-1}$.

An upper bound on S for Theorem 6.5 can be obtained by rewriting (6.15) as

$$\left\{C(x)(x^{s+r}-1)\right\} \neq \left\{U_{s+r}(x)\right\} \tag{6.17}$$

and determining for what values of s+r the above inequality cannot be guaranteed. If (6.17) is compared with (2.15') it is seen that they are identical. It follows that there is no coset which can correct slip when s lies in the range

$$(n-k)/2 \leq s \leq k . \tag{6.18}$$

But, for Fire codes, $n-k = a+b$, where a and b were defined earlier. Hence, (6.18) can be written as

$$(a+b)/2 \leq s \leq k . \tag{6.19}$$

There is also an upper bound on T. Since, in order to correct a burst error of length T or less, it is necessary that

$$\left\{E_{t_1}(x)\right\} \neq \left\{E_{t_2}(x)\right\}$$

or

$$\left\{E_{t_1}(x) + E_{t_2}(x)\right\} \neq 0 \tag{6.20}$$

where $E_{t_1}(x)$ and $E_{t_2}(x)$ are two bursts of length t_1 and t_2 respectively, and $t_1, t_2 \leq T$. Now, the generator polynomial G(x) may be written as

$$\begin{aligned} G(x) &= P(x)(x^a - 1) \\ &= x^a P(x) - P(x) \end{aligned}$$

where P(x) has degree b. Hence G(x), which is itself a Fire code word may be generated by the sum of two bursts, each of length b+1. Therefore, from (6.20) it follows that a Fire code

cannot correct a burst of length b+1 or greater.

Another bound can be obtained from (6.9) which can be written as

$$\left\{C(x)(x^s - 1)\right\} \neq \left\{U_s(x) + E_t(x)\right\}, \; t \leq T, \; s \leq S. \qquad (6.21)$$

The above inequality is the same as (6.7) except for a trivial difference in a sign on the left-hand side. Hence, Theorem 6.3 applies.

The above discussion can be summarized in the following theorem.

Theorem 6.6

No coset of any (n,k) Fire code can correct a burst of length T and S bits of slip, where slip and additive error cannot occur simultaneously, if any of the following conditions are true :

(i) $T \geq b+1$,

(ii) $(a+b)/2 \leq S \leq k$,

(iii) $S + T \geq a+b$.

No special theorem will be given on the ability of Fire codes to correct the simultaneous occurrence of slip and a single burst error. The proof of such a theorem reduces essentially to showing that the sum of three bursts is not a cyclic code word. Fire codes appear to have limited ability for detecting three bursts. Recall that Theorem 5.6 states the performance of cyclic codes in general to correct the simultaneous occurrence of slip and a burst error, in terms of the minimum distance d. This result is of little value for Fire codes as they have small minimum distance. The following lemma serves as a useful reminder of this fact[16]. Recalling the definitions of the parameters in a Fire code given in Section 6.2, we have

Lemma 6.1

Given a Fire code for which $a + \beta < n$, the minimum distance of the code is 4 or less.

Proof

Consider the polynomial

$$(x^{\beta} - 1)(x^{a} - 1) \tag{6.22}$$

where the generator polynomial G(x) of the Fire code is given by

$$G(x) = P(x)(x^{a} - 1). \tag{6.23}$$

P(x) is an irreducible polynomial of degree b and has exponent β. It is desired to show that (6.22) is a Fire code word if $a + \beta < n$, where n is the word length. It is clear that G(x) divides (6.22), since, by definition, P(x) divides $(x^{\beta} - 1)$. It immediately follows that (6.22) is a Fire code word if $a + \beta < n$. By removing the brackets it is seen that (6.22) has a weight of four, hence the Fire code has at least one word of weight four.

Q.E.D.

For binary Fire codes for which $a + \beta < n$, it can in addition be shown that there are no words of weight three. Hence, the minimum weight there is is four.

In Table VI.1, some numerical results for maximum slip S are listed for a few Fire codes, based on the theorems of Chapter VI. In the column for Theorem 6.6, the values listed are for the largest slip S which does not violate any of the three conditions listed in the theorem. Observe that in this chapter, the performance of some of the coset

codes in a noisy channel equals their upper bound. It is believed by the author that Theorems 6.2, 6.3, 6.4, 6.5 and 6.6 are new.

TABLE VI - 1

FIRE CODE				MAXIMUM SLIP(S) FOR THEOREMS LISTED					
(n, k)	b	β	a	T	6.2	6.3*	6.4	6.5	6.6
(42, 22)	6	21	14	1	12	18	12	9	9
				2	11	17	11	9	9
				3	10	16	9	8	9
				4	9	15	7	7	9
				5	8	14	5	5	9
				6	7	13	3	3	9
				7	4	12	0	0	0
				8	4	11	0	0	0
(49, 39)	3	7	7	1	5	8	5	4	4
				2	4	7	4	4	4
				3	3	6	2	2	4
				4	1	5	0	0	0
				5	1	4	0	0	0
(105, 94)	4	15	7	1	5	9	5	4	5
				2	4	8	4	4	5
				3	3	7	2	2	5
				4	2	6	0	0	5
				5	1	5	0	0	0
(105, 84)	6	21	15	1	13	19	13	9	10
				2	12	18	12	9	10
				3	11	17	10	9	10
				4	10	16	8	8	10
				5	9	15	6	6	10
				6	8	14	4	4	10
				7	4	13	0	0	0
				8	4	12	0	0	0
				9	4	11	0	0	0
				10	4	10	0	0	0
(341,325)	5	31	11	1	9	14	9	7	7
				2	8	13	8	7	7
				3	7	12	6	6	7
				5	5	10	2	2	7
				6	3	9	0	0	0
				7	3	8	0	0	0

A Short List of Fire Codes to Illustrate Their Ability to Detect or Correct Slip In The Presence of a Burst Error of Length T or Less.

* Obtained by setting $S + T = a + b - 1$

CHAPTER VII

ANOTHER TECHNIQUE FOR CORRECTING SYNCHRONIZATION ERRORS FOR CYCLIC CODES

7.1 Introduction

Recently, a new technique for correcting the simultaneous occurrence of slip and additive error was introduced by Caldwell[8], and Bose and Caldwell[5]. This technique was later generalized by Weldon[61].

The method essentially involves extending the word length of the cyclic code and restraining some of the information bits, both in a prescribed manner. To illustrate how the word is lengthened, let the n-tuple below be a cyclic code word

$$(a_o, a_1, a_2, \ldots, a_{n-1}) \quad , a_i \in F. \tag{7.1}$$

The above word is now extended as shown in (7.2)

$$(a_{n-L}, \ldots, a_{n-2}, a_{n-1}, a_o, a_1, \ldots, a_{n-1}, a_o, a_1, \ldots, a_R) \tag{7.2}$$

where R and L bits have been added on the right and left respectively, in the manner shown. It is clear that any consecutive n bits selected from (7.2) is a cyclic code word. For the interesting symmetrical case, the number of bits added on either side is equal, i.e., R = L. More relevant to the discussion to follow, is the manner in which the information bits of the cyclic code words are restrained. Recall that any (n,k) cyclic code word can be written in the form

$$W(x) = P(x)\ G(x) \tag{7.3}$$

where G(x) is the generator polynomial and has degree n-k, and P(x) is an arbitrary polynomial of degree less than k.

Following Caldwell[8], restrain P(x) in (7.3) as shown below

$$P(x) = J(x)\, F(x) + 1 \tag{7.4}$$

where F(x) is a polynomial of degree m, and which has exponent $p > R + L$. Also, J(x) is an arbitrary polynomial of degree less than k-m. The new code has k-m information bits and hence q^{k-m} code words, where q is the number of symbols in the code. Each restrained code word can now be written as

$$W(x) = (J(x)\, F(x) + 1)\quad G(x) \ . \tag{7.5}$$

To illustrate the error correcting ability of the method, a theorem given by Weldon[61] is stated and proved below. As elsewhere in this thesis, d is the minimum distance of the cyclic code.

Theorem 7.1

Any (n,k) cyclic code can be extended to form an (n + 2S, k-m) code which can correct the simultaneous occurrence of e or less additive errors and S or less bits of slip, where

$$S = [(q^m - 2)/2] \quad \text{and} \quad e = [(d - 1)/2] \ .$$

Proof

In (7.2), let R = L = S and let F(x) be a primitive polynomial of degree m. Now, assume an extended word derived from (7.5) is transmitted and that $s \leq S$ bits of left

slip and additive error E(x) occur. The receiver then frames the n-tuple

$$x^s W(x) + E(x) , \quad s \leq S. \tag{7.6}$$

The decoder first corrects the additive error by computing the syndrome of (7.6). It decides that additive error is present if

$$\left\{ x^s W(x) + E(x) \right\} \neq 0 \tag{7.7}$$

and since $\left\{ x^s W(x) \right\} = 0$, (7.7) reduces to

$$\left\{ E(x) \right\} \neq 0. \tag{7.8}$$

If $W\left[E(x)\right] \leq e = \left[(d-1)/2\right]$, the decoder can correct the additive error E(x). After correcting E(x), the framed polynomial is

$$x^s W(x) = x^s (J(x) F(x) + 1) G(x) \tag{7.9}$$

which is then divided by F(x) G(x). The remainder after this division is called the synchronization syndrome[61] (or slip syndrome) and in (7.9) this is seen to be the remainder of $\frac{x^s}{F(x)}$. The notation $\left\{P(x)\right\}_F$ will be used to indicate the remainder obtained by dividing P(x) by F(x). As before, $\left\{P(x)\right\}$ with no subscript will mean the remainder after division by the generator polynomial G(x). Applying this notation to (7.9) gives

$$\frac{x^s W(x)}{G(x)} = x^s (J(x) F(x) + 1) \tag{7.10}$$

and

$$\left\{ x^s (J(x) F(x) + 1) \right\}_F = \left\{ x^s \right\}_F \quad . \tag{7.11}$$

The decoder can distinguish between right slip of r bits and left slip of s bits if

$$\left\{ x^s \right\}_F \neq \left\{ x^{-r} \right\}_F \tag{7.12}$$

or

$$\left\{ x^s - x^{-r} \right\}_F \neq 0 \tag{7.13}$$

and the above is equivalent to

$$\left\{ x^{s+r} - 1 \right\}_F \neq 0 \quad . \tag{7.14}$$

Since F(x) is primitive, it does not divide $(x^{s+r} - 1)$ if $s + r < q^m - 1$, hence the decoder can distinguish between right and left slip if $2S < q^m - 1$, or at most, $S = \left[(q^m - 2)/2 \right]$. In addition, the decoder can distinguish between two left slips of p and j bits if

$$\left\{ x^p - x^j \right\} \neq 0, \quad j < p \leq S \, . \tag{7.15}$$

The above is equivalent to

$$\left\{ x^{p-j} - 1 \right\} \neq 0 \tag{7.16}$$

and since $p - j \leq S$, (7.16) is satisfied. A similar proof applies to two right slips, both less than S. Recall that the code words were lengthened by an amount S in each direction, and hence the new word length is $n + 2S$. Also, since F(x) has degree m, the remaining number of information bits is $k - m$.

Q.E.D.

Observe that the slip syndrome of an extended code word is unity, whereas the syndrome of a cyclic code word is zero.

There are two obvious penalties in using the new technique. The first is that the length of the code word is increased by adding redundant bits. Also, when no slip occurs, these extra bits are not used to increase the additive-error-correcting ability of the code. The second penalty is that m of the k information bits are lost. Specifically, for an (n,k) cyclic code, if F(x) has degree m, then from Theorem 7.1, the rate* is reduced from k/n to (k-m)/(n + 2S).

In addition, the decoding process is more complex than for coset codes. To decode a coset code, the decoder computes one syndrome and bases all its decisions on that syndrome. On the other hand, for the codes discussed in this Chapter, there are two syndromes to be computed. They are

(1) the syndrome with respect to G(x) and

(2) the syndrome with respect to F(x).

This tends to lead to a more complex, and hence more expensive decoder.

In the next section, the information bits of the cyclic code will be restrained as described earlier, but the length of the cyclic code word will be unaltered.

---

* The rate may be defined as the number of information bits divided by the length of the code word.

7.2 An Extension of Bose and Caldwell's Technique

Given an (n,k) cyclic code, consider a new code whose code words can be written as

$$W(x) = (J(x)\,F(x) + 1)\;G(x) \tag{7.17}$$

where G(x) is the generator polynomial of the given cyclic code and F(x) is a primitive polynomial of degree m. The essential difference between the code described by (7.17) and the one described by Bose and Caldwell[5] is that no extra bits are added on at the ends of the word in (7.17). The rate for the code given by (7.17) decreases from k/n to (k - m)/n, and there are q^{k-m} code words in this code. Note that (7.17) is still a cyclic code word of the original (n,k) cyclic code, but the new code is not cyclic. In fact it is not even a group code, since the sum of two words in the code does not have the form of (7.17). A number of results will now be derived using the code described in this section. When the primitive polynomial F(x) has degree m, the code in (7.17) will be called an (n, k-m) subset code. Also, in all the results to follow, the minimum distance of the cyclic code will be assumed to be d.

7.3 Detection of Slip by Subset Codes of Cyclic Codes

The theorem which follows concerns the ability of (n, k-m) subset codes to detect slip and additive error.

Theorem 7.2

Given any (n,k) cyclic code, there exists an (n,k-m) subset code having $q^m > n - k + 1$, which can

(1) Detect $n-k$ or less bits of slip if no additive errors are present, and

(2) Detect the simultaneous occurrence of S or less bits of slip and e or less additive errors if $S + e \leq d - 1$.

Proof

Let

$$W(x) = (J(x)\, F(x) + 1)\; G(x) \tag{7.18}$$

be any code word of the subset code, where F(x) is a primitive polynomial of degree m and $q^m > n-k+1$. J(x) is an arbitrary polynomial of degree less than $k-m$.

Part 1

Assume that s bits of left slip occur and that there are no additive errors. As for coset codes, the decoder first computes the syndrome (with respect to G(x)) and will detect an error if

$$\left\{x^s\, W(x) + U_s(x)\right\} \neq 0 \tag{7.19}$$

which reduces to

$$\left\{U_s(x)\right\} \neq 0. \tag{7.20}$$

If $U_s(x) \neq 0$, and $s \leq n-k$, $U_s(x)$ cannot be a cyclic code word and (7.20) will be satisfied. If $U_s(x) = 0$, and $s \leq n-k$, the decoder computes the slip syndrome

$$\left\{x^s(\, J(x)\;\; F(x) + 1)\right\}_F = \left\{x^s\right\}_F \tag{7.21}$$

and the decoder will detect an error if

$$\left\{x^s\right\}_F \neq 1 \tag{7.22}$$

since 1 is the slip syndrome when there is no slip. The previous inequality can be written as

$$\left\{x^s - 1\right\}_F \neq 0 \tag{7.23}$$

and since F(x) is primitive and has degree m, it does not divide $(x^s - 1)$ if $s < q^m - 1$. Since the maximum value of s considered is n - k, choose m such that $q^m - 1 > n - k$, i.e., $q^m > n - k + 1$. A similar proof applies for right slip. This proves the first part of the theorm.

Part 2

Let additive error E(x) and s bits of left slip occur simultaneously. The decoder will detect an error if

$$\left\{x^s\ W(x) + E(x) + U_s(x)\right\} \neq 0. \tag{7.24}$$

The above reduces to

$$\left\{E(x) + U_s(x)\right\} \neq 0 \tag{7.25}$$

which is satisfied if $S + e \leq d - 1$, where $s \leq S$, and $W[E(x)] \leq e$, unless $E(x) + U_s(x) = 0$. If this is the case, the decoder computes

$$\left\{x^s\ (\ J(x)\ F(x) + 1\ )\right\}_F = \left\{x^s\right\}_F \neq 1 \tag{7.26}$$

if $s \leq n - k$. A similar proof applies for right slip.

Q.E.D.

As stated in Theorem 7.2, an (n, k-m) subset code can detect n-k or less bits of slip, whereas an (n,k) coset code can detect only n - k - 1 or less. However, the (n,k) coset code word has m more information bits.

7.4 Correction of Slip by Subset Codes of Cyclic Codes

In the next theorem, the ability of subset codes to correct slip in a noiseless channel will be examined. Let q be the number of symbols (or states) in the code.

Theorem 7.3

Given any (n,k) cyclic code, there exists an (n, k-m) subset code having $q^m > 2[(n-k)/2] + 2$ which can correct all slip $\leq [(n-k)/2]$.

Proof

Assuming p bits of slip occur, the received word has the form

$$x^p W(x) + U_p(x) \tag{7.27}$$

where W(x) is a subset code word, and p is positive for left slip and negative for right slip. As shown in Figure 7.1, the decoder first computes the syndrome of (7.27) to obtain $\{U_p(x)\}$. $U_p(x)$ can be determined from $\{U_p(x)\}$ if*

$$\{U_p(x)\} \neq \{U_r(x)\}, \quad |p|, |r| \leq S. \tag{7.28}$$

The above is satisfied for all distinct $U_p(x)$ and $U_r(x)$ if $S \leq (n-k)/2$. The decoder now subtracts $U_p(x)$ from (7.27) to obtain $x^p W(x)$. The slip syndrome

$$\{x^p W(x)\}_F = \{x^p\}_F \tag{7.29}$$

is now computed. The decoder can determine the magnitude and the direction of the slip p, if

$$\{x^p\}_F \neq \{x^r\}_F, \quad |p|, |r| \leq S$$

---

* $|X|$ means the absolute value of X

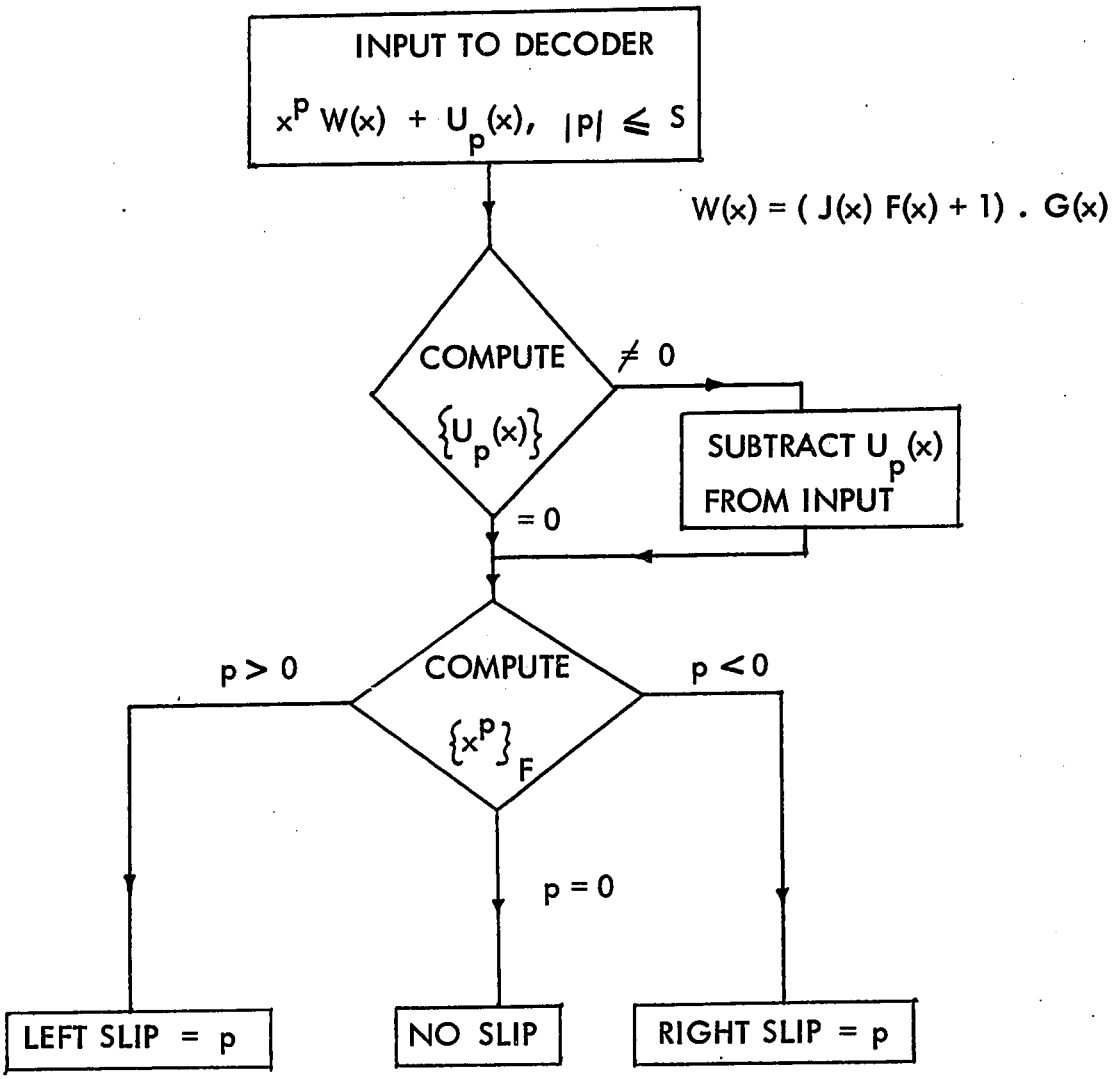

FIGURE 7.1 . FLOW CHART OF DECODING PROCEDURE FOR THEOREM 7.3 .

which can be written as

$$\{x^p - x^r\}_F \neq 0. \tag{7.30}$$

Recalling that F(x) is a primitive polynomial of degree m, (7.30) is satisfied if

$$|p - r| \leq 2S < q^m - 1 .$$

The desired result follows if

$$S \leq [(n-k)/2] \leq (q^m - 2)/2$$

which implies $$q^m \geq 2[(n-k)/2] + 2 .$$

The flow chart in Figure 7.1 outlines the decoding procedure for correcting the slip.

Q.E.D.

Observe that the (n, k-m) subset code corrects $[(n-k)/2]$ or less bits of slip, whereas an (n, k) coset code can correct only $[(n-k-1)/2]$ or less. Furthermore, the subset code can compute the magnitude and the direction of the slip, but for coset codes it was found that slip $\leq [(n-k-2)/2]$. However, as mentioned earlier, the (n, k-m) subset code has m less information bits.

For the channel in the next theorem, assume that slip and additve error do not occur simultaneously in the same n-tuple. The following result can be proved.

Theorem 7.4

Given any (n,k) cyclic code which corrects e additive errors, there exists an (n,k-m) subset code with $q^m > 2e + 1$ which can correct e or less bits of slip and e or less additive errors, if slip and additive error do not occur simultaneously.

Proof

The strategy of the decoder in this theorem is to initially regard all errors as due to additive errors, and correct them as such. It then computes the slip syndrome to see if this assumption was valid. From the slip syndrome the decoder can determine the amount and the direction of the slip, if slip did occur.

Assume first that p bits of slip have occurred. The received n-tuple is then

$$x^p W(x) + U_p(x) \quad , \quad |p| \leq S \tag{7.31}$$

where W(x) is any subset code word. Since $W[U_p(x)] \leq e$ when $|p| \leq e$, the decoder can correct $U_p(x)$ as if it were due to additive error (if $U_p(x) = 0$, the decoder moves immediately to the next step). The remaining n-tuple is now $x^p W(x)$. The next step is to compute the slip syndrome of $x^p W(x)$, which is $\{x^p\}_F$. The decoder can determine the magnitude and the sign of p if $q^m - 1 > 2e$, i.e. if $q^m > 2e + 1$.

If additive error E(x) occurred instead of slip, the received n-tuple would be

$$W(x) + E(x) \tag{7.32}$$

and since $W[E(x)] \leq e$, the decoder can (and does) correct E(x). The decoder now computes the slip syndrome of W(x), which is unity. A slip syndrome of unity tells the decoder that the error is due to additive error, which has already been corrected. If the slip syndrome is not unity, the decoder concludes that slip is present and corrects it as described in the first part of the proof. See Figure 7.2 for the decoding algorithm.

Q.E.D.

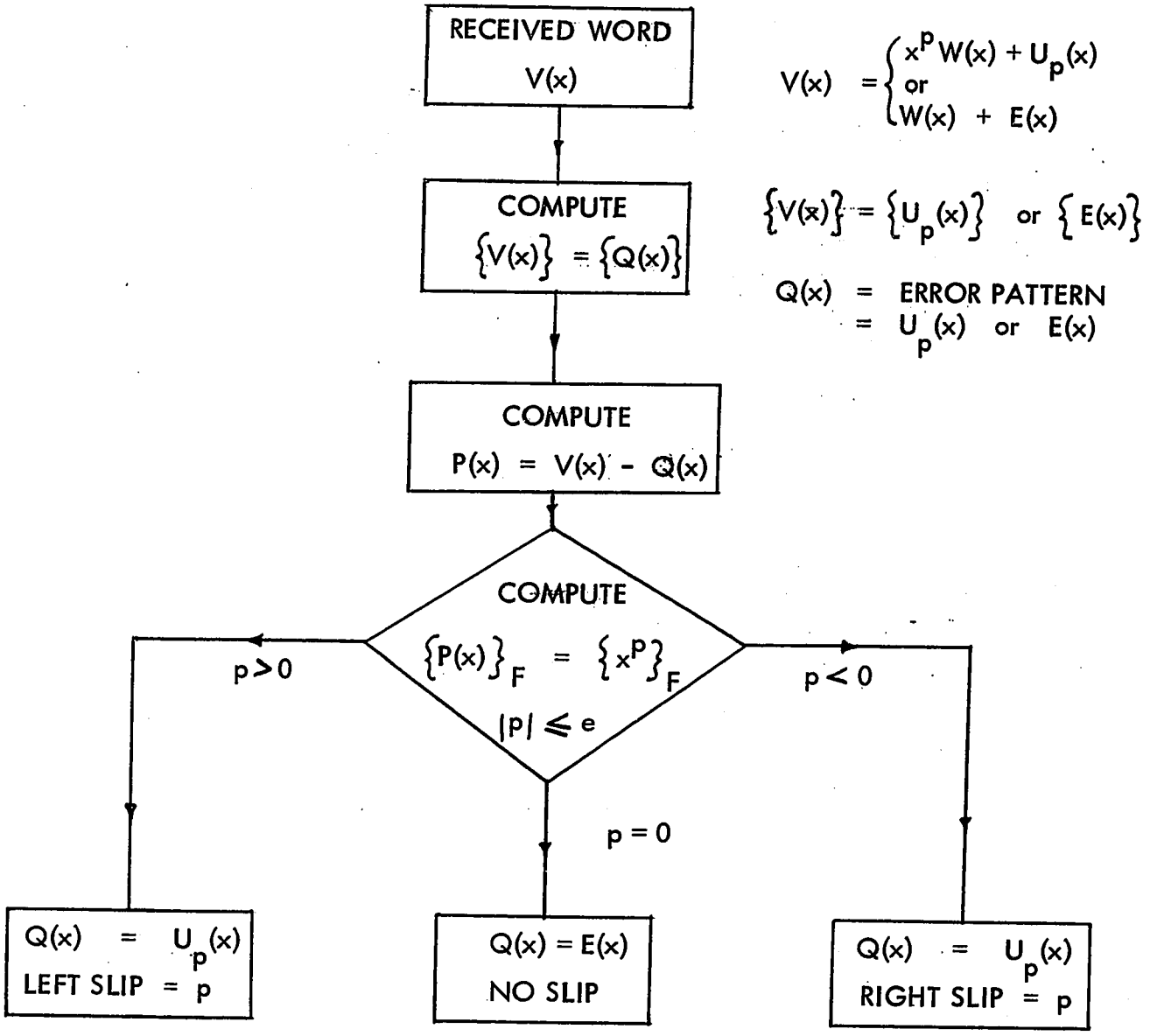

FIGURE 7.2. FLOW CHART FOR DECODER IN THEOREM 7.4.

In Theorem 7.4, note that if $d = 2e + 1$, then the lower bound on m becomes $q^m > d$. In this case, the subset code can correct as many additive errors as the original cyclic code, namely $\lceil (d-1)/2 \rceil$. In addition, the subset code can correct $\lceil (d-1)/2 \rceil$ or less bits of slip if no additive errors occur simultaneously. The penalty is the loss of $\lceil 1 + \log_q d \rceil$ information bits.

7.5 Slip and Additive Error May Occur Simultaneously

In the next theorem, the decoder will correct errors and also detect the presence of slip, when both errors occur simultaneously.

Theorem 7.5

Given an (n,k) cyclic code which corrects e_o additive errors, there exists an (n, k-m) subset code with $q^m > e_o + 1$ which can simultaneously correct e or less additive errors and detect s or less bits of slip if $e + s \leq e_o$.

Proof

The decoder is instructed to initially regard all errors as additive errors and to correct them as such. Suppose that additive error E(x) and p bits of slip occur. The received polynomial is then

$$x^p W(x) + U_p(x) + E(x) \tag{7.33}$$

and if $|p| + e \leq e_o$, where $W[E(x)] = e$, the decoder will correct $U_p(x) + E(x)$ as if it were a pure additive error. The remaining polynomial is $x^p W(x)$ and the decoder now computes the slip syndrome which is $\{x^p\}_F$. The decoder can detect the presence of slip if

$q^m - 1 > e_o \geq p$. Observe that the decoder has corrected the additive error E(x) (and also $U_p(x)$), however, it can only detect that slip is also present. In addition, the decoder knows when slip does not occur, and in this case it corrects e_o or less additive errors. e_o can be chosen as large as $\lceil (d-1)/2 \rceil$.

Q.E.D.

In the next theorem, the decoder can correct slip and additive error, even when they occur simultaneously.

Theorem 7.6

Given any (n,k) cylic code, there exists an (n,k-m) subset code with $q^m > 2\lceil (d-1)/2 \rceil + 1$, which can correct any combination of e additive errors and s bits of slip if $e + s \leq \lceil (d-1)/2 \rceil$.

Proof

As in the previous theorem, the decoder is instructed to initially correct all errors as if they were additive errors. If p bits of slip and e additive errors occur, the received n-tuple is

$$x^p W(x) + U_p(x) + E(x) \tag{7.34}$$

where p is positive for left slip and negative for right slip. If $e + |p| \leq \lceil (d-1)/2 \rceil$, the decoder computes $U_p(x) + E(x)$ and subtracts it from (7.34). It now determines the slip syndrome of $x^p W(x)$ which is $\{x^p\}_F$. However, to determine p, it is necessary that

$$\{x^p\}_F \neq \{x^r\}_F , \qquad p \neq r \tag{7.35}$$

for all $|p|, |r| \leq [(d-1)/2]$. The above inequality can be written as

$$\{x^p - x^r\}_F \neq 0 \tag{7.36}$$

which is equivalent to

$$\{x^{p-r} - 1\}_F \neq 0, \ |p-r| \leq 2[(d-1)/2] . \tag{7.37}$$

Since F(x) is a primitive polynomial of degree m, it does not divide $(x^{p-r} - 1)$ if $q^m - 1 > 2[(d-1)/2] \geq |p-r|$. When m satisfies the previous inequality, it follows from (7.37) that the decoder can determine the magnitude and the direction of the slip.

Q.E.D.

Note that Theorem 7.6 corrects slip, whereas Theorem 7.5 merely detects the presence of slip. However, in Theorem 7.6, more information bits are lost.

7.6 A Variation of the Technique of Subset Codes

The technique discussed in this section involved adding a fixed polynomial C(x) to each code word of the (n, k-m) subset code. Loosely speaking, it combines the techniques of coset codes and subset codes. Each transmitted word of this code will have the form

$$(J(x) F(x) + 1) \ G(x) + C(x) . \tag{7.38}$$

The following theorem will deal with the problem of detection.

Theorem 7.7

Given an (n, k) cyclic code, there exists a code having the form of (7.38) with $q^m \geq d - 1$, which can :

(1) Detect (n - k - 1) or less bits of slip in the absence of additive error.

(2) Detect the simultaneous occurrence of e or less additive errors and S or less bits of slip, where $e + S \leq d - 2$.

Proof

Let $C(x) = 1$.

Part 1

Assume that slip occurs but that no additive errors are present. The code words defined by (7.38) form a subset of the coset code of the given cyclic code. Hence, by Theorem 2.1, the decoder can detect all slip of n - k - 1 bits or less.

Part 2

Assume that p bits of slip and additive error E(x) occur simultaneously. The receiver frames

$$x^p (J(x) F(x) + 1) G(x) + x^p + U_p(x) + E(x) \tag{7.39}$$

where p is positive for left slip and negative for right slip. The decoder will detect an error if (7.39) is not a cyclic code word. If

$$e + |p| + 1 < d \tag{7.40}$$

where $W\left[E(x)\right] = e$, then (7.39) will be a cyclic code word if and only if

$$x^p + U_p(x) + E(x) = 0.$$

When the above is true, (7.39) becomes

$$x^p(J(x)F(x) + 1)\ G(x)\ . \qquad (7.41)$$

The receiver now computes $\{x^p\}_F$, and will detect the presence of slip if $q^m - 1 > |p|$, where $|p| < d - (e + 1)$ from (7.40). The largest non-trivial value of p occurs when $e = 1$, which gives $|p| = d - 3$. This implies that $q^m - 1 > d - 3$ or $q^m \geqslant d - 1$.

Q.E.D.

Theorem 7.7 should be compared with Theorems 2.1 and 7.2. Observe that m is smaller in Theorem 7.7 than in Theorem 7.2, and hence the codes in Theorem 7.7 have more code words. The author believes that all the theorems based on the "(n, k−m) subset codes" of this chapter and Theorem 7.7 are new.

CHAPTER VIII

SUMMARY OF RESULTS

In this Chapter, the results presented in this study will be summarized and an attempt will be made to indicate which of them are original contributions. In some instances, the contribution may consist of a new proof or a stronger theorem*. It will be observed that the thesis has been organized around a set of theorems, which, it is believed, allow easy reference to the results. Although the style is somewhat formal, it should be easy for the reader to find specific results without going through the proofs.

Two techniques were considered for altering cyclic codes to enable them to recover synchronism. In the first, a suitable coset code was formed from the given (n,k) cylic code. Since the coset code has the same distance properties as the parent cyclic code, it also has the same additive-error-detecting and correcting properties. The second method forms an (n, k - m) subset code from the (n,k) cyclic code by restraining m of the k information bits of the cyclic code in a prescribed manner. These subset codes are a variation of the extended cyclic codes developed by Bose and Caldwell. Observe that neither of the techniques employed alters the length of the cyclic code words.

Chapters II to VI, inclusive, are based on the technique of forming coset codes. In Chapter II, slip is assumed to occur in a noiseless channel. The absence of additive errors allows the technique to be introduced without other complicating factors. Hence,

---

* However, to avoid repetition, the policy adopted is that results that are not referenced are believed to be new.

although the noiseless model does not represent real channels, it gives insight into the later study of channels with additive error.

Theorem 2.1, on the ability of coset codes to detect slip, was found independently by Stiffler[52] and Levy[38]. Stiffler uses the vector-matrix representation whereas Levy uses polynomial algebra. The proof given here is similar to Tong's[58]. One difference is that a useful property of cyclic codes is applied which gives an alternative point of view. The property is that no (non-zero) burst of length $n-k$ or less can be an (n,k) cyclic code word. This is a very useful result and is used many times in the thesis. Tong's proof is equivalent, but the above idea is disguised by the albegra. For completeness, his method is also outlined. Later, in Chapter IV, several results on slip are derived in the vector-matrix representation.

Theorem 2.2[52, 58] proves that all (n,k) coset codes are vulnerable to slip exceeding $n-k-1$ bits, which also proves that Theorem 2.1 is optimum. Since the code words are assumed to form a continuous sequence, by symmetry, a slip of s bits in one direction is equivalent to a slip of $n-s$ in the other direction. Hence, if a coset code can detect slip up to one-half a word length i.e. $\lceil n/2 \rceil$, then it can detect all slip. Such a code is called comma-free[24] and Corollary 2.1[38, 52, 58] points out that a coset code is comma-free if and only if $k \leq (n-1)/2$.

The next topic studied is correction of slip in a noiseless channel. The phrase "correction of slip" can have at least two meanings and needs clarification. This matter is taken up in Section 2.3. For instance, Tong[58] uses the phrase "sync-correction capability" to mean the ability of the decoder to distinguish between right slip and left slip. However, this ability does not enable the decoder to determine immediately (i.e. without

searching) the magnitude of the slip. In Theorem 2.3, the decoder can determine the direction of the slip if it does not exceed $(n - k - 1)/2$. Tong[58] comes close to proving this theorem, but omits stating it explicitly. He is content to give his result as an upper bound, and this will be discussed shortly.

Corollary 2.2 states the interesting fact that, depending on the choice of coset in Theorem 2.3, the decoder can determine the magnitude of the slip for one direction only. For instance, if $C(x) = 1$, the decoder can determine the magnitude of the slip for left slip, but cannot do this for right slip. It is the other way around if $C(x) = x^{n-1}$.

Theorem 2.4 gives a coset ($C(x) = 1 + x^{n-1}$) which can compute the magnitude and the direction of the slip if slip does not exceed $(n-k-2)/2$. Comparing Theorems 2.3 and 2.4, it is seen that by sacrificing at most one bit, the decoder can recover sync without having to search.

An upper bound on the ability of coset codes to distinguish between right and left slip in a noiseless channel is given by Theorem 2.5. Tong[58] gives a similar theorem except that the condition $2k \geq n$ is absent from his result. The need for this condition is due to the fact that the random polynomial $U_{s+r}(x)$ is not always completely random. It is recalled that these "random" bits come from an adjacent (n,k) coset code word, which has k information bits (which are arbitrary) and $n-k$ check bits which are determined by the k information bits. This fact imposes restraints on the coefficients of the random polynomial $U_{s+r}(x)$. Theorem 2.5 and Corollary 2.3 were proved with this fact in mind. However, it can be verified that this condition is redundant for all the results on the detection of slip. When $2k \geq n$, Theorem 2.5 proves that Theorem 2.3 is optimum and that Theorem 2.4 is at most one bit short of optimum.

In Chapter III, slip in noisy channels is studied. As might be expected, the analysis becomes appreciably more complicated. In that chapter, the additive noise affects individual transmitted bits independently. In Section 3.2, slip and additive error are assumed not to occur simultaneously, but in Section 3.3 this restriction is removed.

Theorem 3.1, on detection, may be considered to be somewhat obvious, but it is worth stating nevertheless. Theorem 3.2A is very similar to Theorem 4 of Tong[58], but differs in two aspects. First, one redundant condition stated by Tong has been removed. Second, it is shown that the decoder can compute not only the direction, but also the magnitude of the slip. The proof given here is otherwise essentially the same as Tong's, except that the choice of coset is different when the number of additive errors e is even (see (3.9a)). This fact serves as a reminder that the choice of coset is often not unique. Theorems 3.2B and 3.2C are new theorems which correct larger values of slip than Theorem 3.2A when d is large and e is small. Unlike 3.2A, they are valid only for binary codes.

Theorem 3.3A is on the detection of the simultaneous occurrence of slip and additive errors by coset codes. This theorem is suggested by Tong's work[58] but he does not state it explicitly. Theorems 3.3B and 3.3C, valid only for binary codes, are new and give larger values of correctable slip when d is large and e is small.

It may sometimes be preferred to correct additive errors when they occur without slip, and merely detect the simultaneous occurrence of slip and additive error. This situation is covered in Theorem 3.4. Note that in Theorem 3.4 the decoder must distinguish between pure additive error and the simultaneous occurrence of slip and additive error. Corollary 3.1 adds some more flexibility to Theorem 3.4.

Theorem 3.5 examines the quite general case of the correction of both slip and additive error, when they occur simultaneously. Observe that this is achieved, as for all coset codes discussed in this study, without increasing the length of the code word or sacrificing any of the information bits. However, the available redundancy in the code word has been divided between additive error correction and slip error correction. Consequently the maximum number of additive errors that can be corrected has been reduced.

The decoder in Theorem 3.5 can compute the magnitude of the slip for left slip, but not for right slip (additive errors are present in both cases). For the reduction of at most one bit in S, the decoder in Theorem 3.6 can compute the magnitude and direction of the slip, when slip and additive error occur simultaneously. A bound on the performance of Theorems 3.5 and 3.6 is provided by the upper bound on S given in Theorem 3.7. However, since this bound may not be tight, it should not be used to draw unfavourable conclusions about Theorems 3.5 and 3.6.

In Chapter IV the vector-matrix description of cyclic codes is taken up, in contrast to the polynomial description which had been used earlier. As far as the author is aware, Stiffler[52] was the first to use coset codes and the vector-matrix approach in studying the problem of synchronization of cyclic codes. However, his theorems only considered the detection of slip when additive errors cannot occur simultaneously. In Chapter IV, the vector-matrix approach has been extended to handle various combinations of slip and additive error. To avoid undue repetition only a representative selection of the previous theorems are re-examined. A fairly complete development has been given elsewhere[55]. In particular, the method has been extended to handle correction of slip in the noiseless channel, and further, correction of the simultaneous occurrence of slip and additive error

for the noisy channel. It is then a fairly straightforward matter to apply the vector-matrix method to other results not explicitly covered here.

One motive for using the vector-matrix representation is that many communication engineers may be familiar with the essentials of matrix theory but not with the concepts and terminology of modern algebra. However, even for readers familiar with both subjects, the two approaches may result in a better understanding.

Lemma 4.1 is given as Theorem 8.2 by Peterson[39] who gives an algebraic proof. The proof given here, using matrix theory was developed by the author, who has not seen a similar one elsewhere. Lemma 4.2 may be known, but the author has not seen it in the literature. However, the fact that there are q^{n-k} distinct syndromes for an (n,k) cyclic code with q symbols is well known.

In the previous chapters, the additive errors were assumed to affect the transmitted bits independently. However in Chapter V, the additive errors are assumed to occur in bursts. The class of codes used is the coset codes of binary cyclic codes. A special kind of burst error is the multiple-adjacent-additive error, which is simply a sequence of consecutive errors. Theorem 5.1 covers the detection of the simultaneous occurrence of slip and a single burst of adjacent-additive-errors; Theorem 5.2 deals with the correction of both such errors, assuming that they cannot occur simultaneously. Theorem 5.3 examines the detection of the simultaneous occurrence of slip and a single burst (in the general meaning of "burst"). Two useful lemmas, 5.1 and 5.2, are then given, as they assist in the proof of later theorems.

In Section 5.4, Theorem 5.4 deals with the correction of slip and the detection of a single burst, assuming that the two types of error do not occur simultaneously.

In Theorem 5.5, both types of error are corrected under the same assumptions. For the more general channel where slip and burst errors occur simultaneously, the decoder in Theorem 5.6 corrects both slip and a single burst. Finally, Theorem 5.7 sets an upper bound on the ability of a coset code to correct both slip and a single burst when they occur simultaneously. It will be noticed that there is a significant gap between the performance of Theorem 5.6 and the bound of Theorem 5.7. However, it should be kept in mind that the only parameter used in proving this theorem and nearly all the others, is the minimum distance d of the (n,k) cyclic code. To bring the performance achieved closer to the bound, it may be necessary to restrict the discussion to a subset of the cyclic codes, for example the BCH codes. This would be equivalent to specifying more parameters.

Another important subset of the class of cyclic codes are the Fire codes[16, 39]. They are studied in Chapter VI and the cosets of these codes usually give results closer to the upper bounds. Theorems 6.2 to 6.6 examine the ability of coset codes of Fire codes to handle a variety of combinations of slip and burst errors. Lemma 6.1 serves as a reminder of the fact that Fire codes have small minimum distance and are not suitable for channels with independent additive errors[16].

Chapter VII introduces the other technique for altering cyclic codes to enable them to detect and correct slip. This technique is adapted from the the one developed by Bose and Caldwell[5]. In Chapter VII, the main ideas of Bose and Caldwell's method are briefly outlined and Theorem 7.1[61] is proved to make the ideas concrete and to give some quantitative results. From Theorem 7.1 it is seen that the extended code can correct the simultaneous occurrence of slip and additive error without loss of any additive-error-correcting ability. The penalty is that

(1) the length of the code word is increased by the addition of redundant bits and

(2) the number of information bits in the original cyclic code is reduced.

The extension of Caldwell's method studied here is to restrain some of the information bits in the same manner but not to add any bits to the ends of the code word. Starting with an (n,k) cyclic code, the new codes, called (n, k-m) subset codes, have k-m information bits and word length n. The various lower bounds on m are given in Chapter VII.

Theorem 7.2 describes the ability of the (n,k-m) subset codes to detect slip and additive errors. Theorem 7.2 can be compared with Theorem 3.3A. Observe that in order to detect the simultaneous occurrence of slip S and additive error e, Theorem 7.2 requires that $S + e \leq d - 1$ (necessary and sufficient) whereas Theorem 3.3A requires that $S + 2e \leq d - 2$ (necessary but not sufficient), for the coset code. However, in Theorem 7.2, m information bits are lost, where $q^m > n - k + 1$.

In Theorem 7.3, the subset code corrects slip in a noiseless channel and in Theorem 7.4 the subset code corrects slip and additive error, in a channel where slip and additive error do not occur simultaneously. Slip and additive error can occur simultaneously in Theorems 7.5 and 7.6. In Theorem 7.5 the decoder corrects additive errors but only detects slip. However, in Theorem 7.6 the decoder corrects both types of error. It is interesting to compare Theorem 7.6 with Theorem 7.1 which uses Bose and Caldwell's technique. In both theorems the decoder can correct slip and additive error even when they occur simultaneously. However, 7.1 can correct the maximum number of additive

errors, i.e. $\lceil (d-1)/2 \rceil$, when slip is present, whereas in 7.6 the number of additive errors that can be corrected is reduced by the amount of slip present. This is offset by the fact that the code words are longer in Theorem 7.1. The last result is given in Theorem 7.7, and is a combination of the two previous techniques. The code words of this code can be obtained by forming the (n, k-m) subset code and then generating a " coset " of this code by adding C(x). This combined technique appears to have some advantages for detection schemes. It is instructive to compare the slip detecting capabilities of Theorem 7.7 with Theorems 2.1 and 7.2. Note that although Theorem 7.2 detects one more bit of slip than Theorem 7.7, the code in Theorem 7.2 sacrifices more information bits.

In general, it is not easy to choose simple criteria on which to compare the performance of codes in an environment of slip and additive error. One problem is assigning the relative importance of slip errors and additive errors. In fact, slip errors may raise important theoretical problems in information theory[25], since Shannon's development does not take synchronization errors into account.

CHAPTER IX

CONCLUSIONS

The problem of detecting and correcting loss of synchronization, or slip, for cyclic codes has been examined for both noiseless and noisy channels. Two different techniques which reduce the vulnerability of cyclic codes to slip errors were considered. One technique was to select a suitable coset code of the given (n,k) cyclic code by adding a fixed n-tuple to each cyclic code word before Transmission. The other was to generate an " (n, k-m) subset code " by restraining m of the k information symbols of the cyclic code, in a prescribed manner. The techniques did not alter the length of the cyclic code words or reduce their additive-error-detecting capability. To give two points of view, both the vector-matrix and the polynomial algebra representation of cyclic codes were utilized.

Several new theorems were presented on the ability of coset codes of cyclic codes to recover synchronism. It was shown that by the proper choice of coset, the decoder could determine both the magnitude and the direction of the slip, which eliminated the necessity of searching for synchronism. For the noisy channel, both independent additive errors and burst errors were considered. In particular, a class of coset codes was described which could correct both slip and additive error, even when they occurred simultaneously. In addition, it was shown that there exist coset codes of Fire codes which could detect and correct slip in a channel with burst errors.

A class of codes called " (n, k-m) subset codes " was described, which were adapted from the extended cyclic codes of Bose and Caldwell. However, unlike the extended codes, the subset codes have the same word length as the original cyclic code. It was demonstrated that these subset codes have the ability to recover synchronism, even

in the simultaneous presence of additive error.

In this study, attention has been concentrated on situations where the decoder can detect or correct slip with certainty. A suitable area for future investigation could be the probability of undetected error when slip and additive error lie outside the guaranteed range. It may also be possible to improve on the performance of the theorems presented or, if this is not possible, to tighten the upper bounds. However, this may require the specification of more parameters of the cyclic codes than was required in this study.

APPENDIX I

It is desired to prove that

$$W\left[Q(x)\,(x^i + 1)\right] \leq 2 + 3e + 2S - 2(s - m) \tag{1A}$$

where $i = S - s + 1$ and

$$Q(x) = D_s(x) + U_s(x) + E(x), \quad 0 < s \leq S \tag{2A}$$

and

$$W\left[U_s(x) + E(x)\right] = s + e - m \tag{3A}$$

and

$$W\left[E(x)\right] \leq e \ . \tag{4A}$$

Proof

The worst case occurs when $W\left[E(x)\right] = e$ which gives

$$W\left[E(x)\,(x^i + 1)\right] \leq 2e \ . \tag{5A}$$

Subtracting (5A) from (1A), it is sufficient to prove that

$$W\left[R(x)\,(x^i + 1)\right] \leq 2 + e + 2S - 2(s - m) \tag{6A}$$

where

$$R(x) = D_s(x) + U_s(x) \ . \tag{7A}$$

When $W\left[E(x)\right] = e$, it follows from (3A) that

$$W\left[U_s(x)\right] = s - m \tag{8A}$$

which means that m of the possible s terms of $U_s(x)$ are missing.

When e is even, recall that

$$C(x) = \sum_{j=0}^{e/2} x^{j(S+1)} \tag{9A}$$

and hence

$$D_s(x) = C(x)(x^s + 1) - 1$$

$$= x^s \sum_{i=0}^{e/2} x^{i(S+1)} + \sum_{k=1}^{e/2} x^{k(S+1)}$$

$$= x^s + \sum_{k=1}^{e/2} x^{k(S+1)} (x^s + 1). \qquad (10A)$$

Substituting (10A) into (7A), it can be shown that, after making the necessary cancellation of terms, the inequality (6A) is satisfied.

APPENDIX II

A SIMPLE INEQUALITY RELATING MINIMUM DISTANCE AND THE NUMBER OF PARITY CHECK BITS FOR BINARY CYCLIC CODES *

Theorem

For any (n, k) binary cyclic code, with minimum distance d, the following inequality is true :

$$d \leq 2/3 \left[(n - k) + 2\right].$$

Proof

Without loss of generality we assume that the k information bits occupy the k left positions of the code word as shown below :

$$(a_1, a_2, \ldots, a_k, b_1, b_2, \ldots b_{n-k})$$

where the a_i's are information bits and the b_i's are check bits.

Consider the code word whose information bits are all zero except the one in the k'th position, i.e. $a_k \neq 0$. Let this word have p zeros among its parity check bits. Then its weight is $1 + (n-k) - p$ and hence

$$d \leq n - k + 1 - p, \quad p \geq 0. \tag{1B}$$

Now cyclically shift this word one bit and add the shifted version to the original word to obtain a code word whose weight is at most $2p + 2$. This implies

$$d \leq 2p + 2 \tag{2B}$$

---

* See also the more general inquality in : Solomon, G. and J.J. Stiffler, "Algebraically Punctured Cyclic Codes", Information and Control, Vol. 8, No. 2, pp. 170-179, 1965.

and from (1B) we have

$$p \leq n - k + 1 - d. \tag{3B}$$

Substitute (3B) into (2B) to obtain

$$d \leq 2(n - k + 1 - d) + 2$$

or

$$d \leq 2/3\,(n - k + 2) \quad .$$

Q.E.D.

BIBLIOGRAPHY

1. Alexander, A.A., R.M. Gryb and D.W. Nast, "Capabilities of the Telephone Network for Data Transmission", Bell System Tech. J., Vol. 39, pp. 431 - 476, May 1960.

2. Ash, R.B., "Information Theory", J. Wiley and Sons, 1965.

3. Barker, R.H., "Group Synchronizing of Binary Digital Systems", in Communication Theory, W. Jackson Ed., Academic Press, N.Y., 1953.

4. Birkhoff, G. and S. Maclane, "A Survey of Modern Algebra", (Third Edition), The Macmillan Co., 1965.

5. Bose, R.C., and J.G. Caldwell, "Synchronizable Error-Correcting Codes", Information and Control, Vol. 10, No. 6, pp. 616-630, June 1967.

6. Bose, R.C. and D.K. Ray-Chaudhuri, "On a Class of Error-Correcting Binary Group Codes", Information and Control, Vol. 3, pp. 68-79, March 1960.

7. Bose, R.C. and D.K. Ray-Chaudhuri, "Further Results on Error Correcting Binary Group Codes", Information and Control, Vol. 3, pp. 279-290, Sept. 1960.

8. Caldwell, J.G., "Synchronizable Error-Correcting Codes", Doctoral Dissertation, Department of Statistics, University of North Carolina, Chapel Hill, May 1966.

9. Carmichael, R.D., "Introduction to the Theory of Groups of Finite Order", Ginn and Company, 1937, also Dover Publications, Inc., 1956.

10. Crick, F.H.C., J.S. Griffith and L.E. Orgel, "Codes Without Commas", Proceedings of the National Academy of Sciences of the United States, Vol. 43, pp. 416-421, 1957.

11. Eastman, W.L., "On the Construction of Comma-Free Codes", IEEE Trans. on Information Theory, Vol. IT-11, pp. 263-267, April 1965.

12. Eastman, W.L. and S. Even, "Some Further Results on Synchronizable Block Codes", IEEE Trans. on Information Theory, Vol. IT-12, No. 3, pp. 404-406, July 1966.

13. Eastman, W.L. and S. Even, "On Synchronizable and PSK-Synchronizable Block Codes", IEEE Trans. on Information Theory, Vol. IT-10, pp. 351-357, Oct. 1964.

14. Elias, P., "Coding for Noisy Channels", IRE Convention Record, Part 4, pp. 37-45, 1955 National Convention, March 1955.

15. Fano, R.M., "Transmission of Information", The MIT Press, 1961.

16. Fire, P., "A Case of Multiple-Error-Correcting Binary Codes for Non-Independent Errors", Sylvania Report RSL-E-2, Mt. View, California, March 1959.

17. Fontaine, A.B. and R.G. Gallager, "Error Statistics and Coding for Binary Transmission over Telephone Circuits", Proceedings of the IRE, Vol. 49, pp. 1059-1065, June 1961.

18. Frey, A.H., "Message Framing and Error Control", IEEE Trans. on Military Electronics, Vol. MIL-9, pp. 143-147, April 1965.

19. Gilbert, E.N., "A Comparison of Signaling Alphabets", Bell System Tech. J., Vol. 41, pp. 504-522, May 1962.

20. Gilbert, E.N., "Information Theory After 18 Years", Science, Vol. 152, No. 3720, pp. 320-326, April 1966.

21. Gilbert, E.N., "Cyclically Permutable Error-Correcting Codes", IEEE Trans. on Information Theory, Vol. IT-9, pp. 175-182, July 1963.

22. Gilbert, E.N., "Synchronization of Binary Messages", IRE Trans. On Information Theory, Vol. IT-6, pp. 470-477.

23. Golomb, S.W. and B. Gordon, "Codes with Bounded Synchronization Delay", Information and Control, Vol. 8, pp. 355-372, 1965.

24. Golomb, S.W., B. Gordon and L.R. Welch, "Comma-Free Codes", Canadian Journal of Mathematics, Vol. 10, pp. 202-209, 1958.

25. Golomb, S.W., J.R. Davey, I.S. Reed, H.L. Van Trees and J.J. Stiffler, "Synchronization", IEEE Trans. on Communications Systems, Vol. CS-11, pp. 481-491, December 1963.

26. Golomb, S.W., "Shift Register Sequences", Holden-Day, Inc., 1967.

27. Golomb, S.W., L.R. Welch and M. Delbruck, "Contruction and Properties of Comma-Free Codes", Biol. Med. Danske Vid. Selsk, Vol. 23, pp. 1-34, No. 9, 1958.

28. Hackett, Jr., C.M. "Synchronization of Cyclic Codes", First Annual Conference on Information Sciences and Systems, Princeton, New Jersey, March 1967.

29. Hamming, R.W., "Error Detecting and Error Correcting Codes", Bell System Tech. J., Vol. 29, pp. 147-160, April 1950.

30. Hocquenghem, A., "Codes Correcteurs d'erreurs", Chiffres, Vol. 2, pp. 147-156, September 1959.

31. Jaynes, E.T., "Note on Unique Decipherability, IRE Trans. on Information Theory,

pp. 98-102, September 1959.

32. Jiggs, B.H., "Recent Results in Comma-Free Codes", Can. J. Math., Vol. 15, pp. 178-187, 1963.

33. Kasahara, Y. and M. Kasahara, "Notes on Synchronization for Burst Error Correction Code", Journal of the Institute of Electrical Communications Engineers of Japan, Vol. 47, pp. 85-95, June 1964.

34. Kasahara, Y. and M. Kasahara, "Synchronization", Journal of the institute of Communications Engineers of Japan, Vol. 49, pp. 33-40, 1966.

35. Kendall, W.B. and I.S. Reed, "Path - Invariant Comma-Free Codes", IRE Trans. on Information Theory, Vol. IT-8, pp. 350-355, October 1962.

36. Lee, C.Y., "Some Properties of Non-Binary Error-Correcting Codes", IRE Trans. on Information Theory, Vol. IT-4, pp. 77-82, 1958.

37. Levinstein, V.I., "Binary Codes for the Correction of Insertions, Deletions and Changes of Symbol", Dokl. Akad. Nauk. Ukr. S.S.R., Vol. 163, pp. 845-848, 1965.

38. Levy, J.E., "Self-Synchronizing Codes Derived from Binary Cyclic Codes", IEEE Trans. on Information Theory, Vol. IT-12, No. 3, pp. 286-290, July 1966.

39. Peterson, W.W., "Error-Correcting Codes", MIT Press, 1961.

40. Peterson, W.W., "Error-Correcting Codes : A Discussion of Status", Progress in Radio Science, Vol. 6, 1960-1963, ed. F.L.H.M. Stumpers, Elsevier Co., 1966.

41. Peterson, W.W. and J. Massey, "Coding Theory", IEEE Trans. on Information Theory, Vol. IT-9, pp. 223-229, October 1963.

42. Posner, E.C., "Simultaneous Error Correcting and Burst Error Detection Using Binary Linear Cyclic Codes", J. Soc. Indust. and Appl. Math., Vol. 13, pp. 1087-1095, 1965.

43. Prange, E., "Cyclic Error-Correcting Codes in Two Symbols", AFCRC-TN-57-103, Air Force Cambridge Research Center, Cambridge, Mass., September 1957.

44. Ramamoorthy, C.V. and D.W. Tufts, "Reinforced Prefixed Comma-Free Codes", IEEE Trans. on Information Theory, Vol. IT-13, No. 3, pp. 366-371, July 1967.

45. Scholtz, R.A., "Codes with Synchronization Capability", IEEE Trans. On Information Theory, Vol. IT-12, pp. 135-140, April 1966.

46. Schutzenberger, M.P., "On Synchronizing Prefix Codes", Information and Control, Vol. 11, No. 4, pp. 396-401, October 1967.

47. Sekimoto, T. and H. Kaneko, "Group Synchronization for Digital Transmission Systems", IRE Trans. on Communications Systems, Vol. CS-10, pp. 381-390, December 1962.

48. Sellers, F.F., "Bit Loss and Gain Correction Code", IEEE Trans. on Information Theory, Vol. IT-8, pp. 35-38, January 1962.

49. Shannon, C.E., "A Mathematical Theory of Communication" Bell System Tech. J., Vol. 27, pp. 379-423 and 623-656, 1948.

50. Slepian, D., "A Class of Binary Signalling Alphabets", Bell System Tech. J., Vol. 35,

pp. 203-234, January 1956.

51. Slepian, D., "Some Further Theory of Group Codes", Bell System Tech. J., Vol. 39, pp. 1219-1252, September1960

52. Stiffler, J.J., "Comma-Free Error Correcting Codes", IEEE Trans. on Information Theory, Vol. IT-11, pp. 107-112, January 1965.

53. Stiffler, J.J., "Synchronization Methods for Block Codes", IRE Trans. on Information Theory, Vol. IT-8, pp. 525-534, September 1962.

54. Stiffler, J.J., "Synchronization Techniques", in Digital Communications with Space Applications, by S. Golomb, ed., Prentice-Hall, New Jersey, 1964.

55. Tavares, S.E. and M. Fukada, "Matrix Approach to Synchronization Recoveryfor Binary Cyclic Codes", to be published in the January 1969 issue of the IEEE Transactions on Information Theory.

56. Tavares, S.E. and M. Fukada, "Detection and Correction of Synchronization Error in the Presence of Additive Error for Binary Cylic Codes", Presented at the Second Annual Princeton Conference on Information Sciences and Systems, March, 1968.

57. Tavares, S.E. and M. Fukada, "Synchronization of Cylic Codes in the Presence of Burst Errors", to be presented at the 1968 Canadian Symposium on Communications, Montreal, November 7-8.

58. Tong, S.Y., "Synchronization Recovery Techniques for Binary Cyclic Codes", Bell System Tech. J., Vol. 45, pp. 561-596, April 1966.

59. Ullman, J.D., "Near-Optimal, Single-Synchronization Error-Correcting Code", IEEE Trans. on Information Theory, Vol. IT-12, No. 4, pp. 418-424, October 1966.

60. Ullman, J.D., "On the Capabilities of Codes to Correct Synchronization Errors", IEEE Trans. on Information Theory, Vol. IT-13, No. 1, pp. 95-105, January 1967.

61. Weldon, Jr., E.J., "A Note on Synchronization Recovery with Extended Cyclic Codes", Presented at First Annual Conference on Information Sciences and Systems, Princeton, New Jersey, March 1967.