SYNCHRONIZATION OF BINARY CYCLIC CODES

by

Gerald Seguin

Submitted to the Department of Electrical Engineering in partial fulfilment of the requirements for the degree

of

Master of Science
Department of Electrical Engineering
Faculty of Pure and Applied Science
University of Ottawa
OTTAWA, ONTARIO

AUGUST, 1969

ABSTRACT

In a communication system if an (n,k) error-correcting block code is used, then at the receiver the incoming sequence is only meaningful when considered in blocks of length n. Hence one of the tasks of the receiver is to partition the incoming sequence into blocks of length n where each block corresponds to the transmitted one which caused it. However due to noise in the system, this partitioning process will sometimes be done incorrectly, in which case a slip (or synchronization) error is said to have occurred.

A code which can correct both additive and slip errors is called a synchronizable error correcting code.

The purpose of the thesis is to present a class of synchronizable error correcting codes. The following result has been established. Let V be on (n,k) cyclic code which can simultaneously correct one burst of length S or less always occurring in the first 25 places, a burst of length S or less always occurring in the last 25 places and a third error $\epsilon(x)$ occurring elseqhere. Let V' be the shortened ($n' = n-2S, k-25-1$), $k \geq 25+1$, code obtained from V and consisting of only those words which start with a 1. Then V' is on $\epsilon(x)$ synchronizable error correcting code capable of correcting the simultaneous occurrence of $\epsilon(x)$ additive errors and S slip errors. Modifications of this Theorem will also be presented. The results obtained will then be applied to existing error-correcting codes.

ACKNOWLEDGEMENTS

The author would like to thank Professor S.G.S. Shiva for his guidance, patience and encouragement throughout this research.

Thanks are also due to the Electrical Engineering Department of the University of Ottawa, Northern Electric and the National Research Council for financial assistance.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1. Coding For Error Control

In communications we are dealing with the problem of transmitting information from a point A to a point B. For example A could be the memory unit and B the arithmetic unit in a modern digital computer. At the other end of the spectrum, A could be a weather satellite orbiting the earth and B a receiving ground station. What lies between A and B need not be the conventional type of channel we usually imagine but could be a storage device. This latter type of system is common in instrumentation where information is often recorded on magnetic tape and later retrieved from the tape and processed. We have the same situation in data centers where the information is stored on magnetic tape or discs.

Since the advent of the digital computer, digital techniques have become very popular. Because the computer processes information in a digital form it seems logical to record and even transmit information in such a form. A system which transmits information in a digital format is termed a "digital data transmission system". A typical system of this form is shown in figure 1.

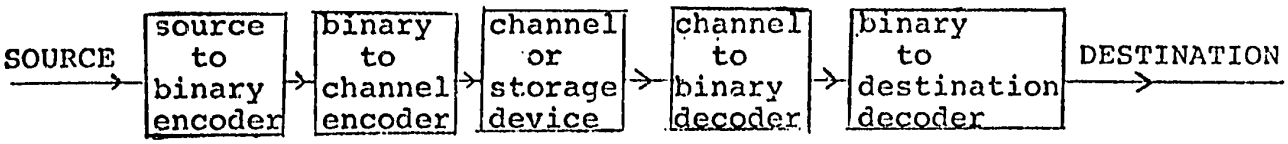


FIGURE 1

In this thesis we will be concerned solely with this type of system.

In Figure 1, the'source to binary encoder' converts the source signal (e.g. a voltage or current function) into a binary sequence (i.e. a sequence of zeros and ones). Next the 'binary to channel encoder' converts this binary sequence into a form adaptable to the channel. This last step is usually called "modulation". For example a 0 could be transmitted as a pulse of frequency f_0 and a '1' as a pulse of frequency $f_1 \neq f_0$. This in a sense would correspond to frequency modulation in conventional radio transmission. Various other methods are used such as amplitude modulation, phase modulation, etc...

In such a system, if we transmit a '0' or a '1' then we expect to receive a '0' or a '1' respectively. Occasionally, due to noise in the channel (or recording failures in

the case of a storage device), a transmitted 0 will be incorrectly interpreted as a 1 and vice-versa. If one wants to minimize such incorrect decisions then the first step might be to improve the channel. This, however, is limited and in some cases we have no access to the channel. If further reliability is required, then we can use error-correcting codes. Such high reliability is often required in cases where retransmission is impossible (weather satelite) or where errors cause further errors such as computation in a digital computer.

A digital data transmission system using error-correcting codes would then be as in Figure 2.

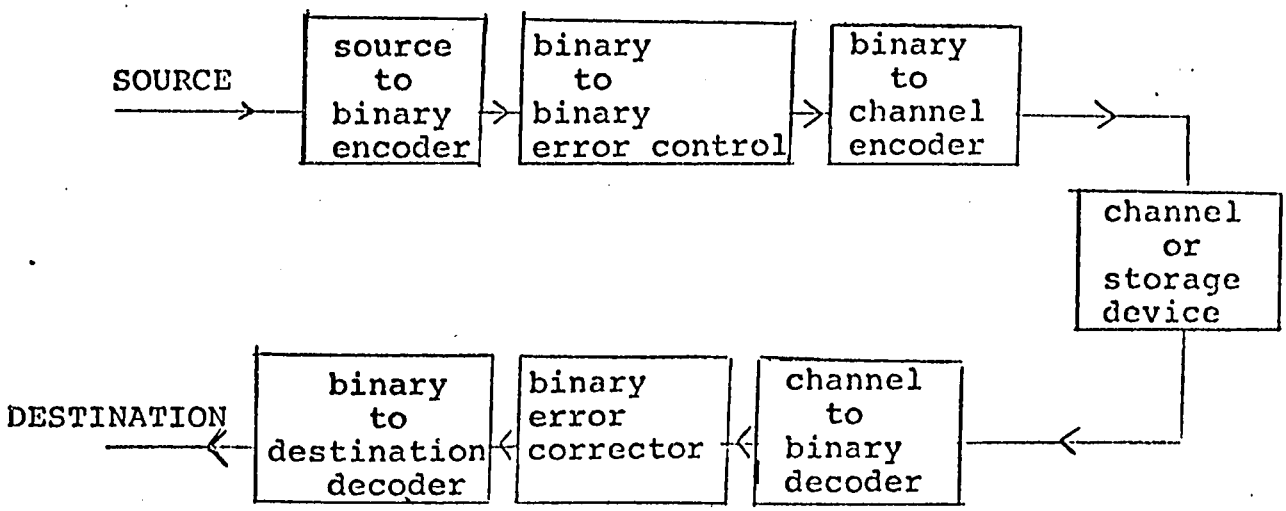


Figure 2

The basic idea is very simple. We take a set of k message digits which we wish to transmit, annex to them r check digits, and transmit the entire n=r+k digits. Assuming that the channel noise changes sufficiently few of these transmitted digits, the r check digits may provide enough information to the receiver to enable it to detect and correct the errors. It is clear that these r check digits cannot be chosen arbitrarily but they must be a function of the k information digits.

One of the first error correcting code to come along is the so called parity check code. For example if we wish to transmit the k-tuple $(a_1\ a_2\ ..\ a_k)$ where a_i=0 or 1, then we would transmit $(a_1\ a_2\ ..\ a_k\ a_{k+1}\ )$ where the check bit $a_{k+1}=\sum_{i=1}^{k} a_i$ sum modulo 2. This is equivalent to making a_{k+1}=0 if the number of non-zero information bits is even and a_{k+1}=1 in the other case. Hence this type of code can detect all error patterns of odd weight. At this point we will make a few definitions which will facilitate the remaining discussion.

DEFINITION 1.1.1.

If $(a_1\ a_2\ ...\ a_n)$ and $(b_1\ ..\ b_n)$ are two binary n-tuples then their modulo-2 sum is a third n-tuple $(c_1..c_n)$ where

$c_i=a_i+b_i$ and $+$ 0 stands for modulo 2 addition defined by; $1\oplus1=0$, $1\oplus0=0\oplus1=1$, $0\oplus0=0$.

DEFINITION 1.1.2.

Let $V=(a_1\ldots a_n)$ be a binary n-tuple. The weight of V, denoted W(V), is the number of non-zero coeficients in V. e.g. $W\{V=(0110101)\}=4$.

DEFINITION 1.1.3.

If V_1 and V_2 are two binary n-tuples, the Hamming [4] distance between them is $W(V_3)$ where $V_3=V_1\oplus V_2$.

EXAMPLE 1.1.1.

If $V_1=(1011011)$, $V_2=(0111010)$ then the Hamming distance between V_1 and V_2 is 3 since $V_3=(1100001)$ and $W(V_3)=3$.

If we transmit an n-tuple $V=(a_1\ldots a_n)$ over a noisy channel and we receive $R=(b_1\ldots b_n)$ then it follows that the coefficients of V which have been changed by noise are the non-zero coefficients of $E=V\oplus R$.

DEFINITION 1.1.4.

If we transmit $V=(a_1\ldots a_n)$ and receive $R=(b_1\ldots b_n)$ then the error vector is $E=R\oplus V$.

Hence it follows that if we know what error has occured during transmission then we know the transmitted code word as;

$$V=R \oplus E$$

The addition of a parity check to the k information bits will enable the decoder to detect all errors of odd weight (or even weight if $a_{k+1}=\left(\sum_{i=1}^{K} a_i\right)+1$, but cannot correct such errors. In 1950 Hamming [9] developed a class of single error correcting (SEC), double error detecting codes. He also developed a relation between the error correcting capability of a code and the concept of Hamming distance between code words. More specifically if V is a code, then V can correct all error patterns of weight e or less if and only if the Hamming distance between any two code words is d or more where

$$d=2e+1. \qquad 1.1.1.$$

The SEC Hamming codes fall into the class of linear codes which we define as:

DEFINITION 1.1.5.

Let H be an r by n matrix whose entries are 0's or 1's. The set V of all n-tuples v satisfying.

$$H v^T = 0 \qquad 1.1.2.$$

is a linear code. v^T stands for the transpose of v and H is called the parity check matrix for V.

We also have,

DEFINITION 1.1.6.

If V is a linear code with parity check matrix H, then for any binary n-tuple w

$H w^T$ is called the

syndrome of w.

If all the columns of H are distinct and non-zero then the code associated with H is a SEC Hamming code [1,2] . In general if all sets of 2t or less columns of H form a linearly independent set, then V is a t error correcting code [1] .

Associated with the H matrix is a k by n matrix G called the generator matrix of the code [1,2,3,] . Since a linear code V is a vector space over GF(2)={0,1} then we can say that the rows of G form a basis for V[1,2,3,5] . If we wish to transmit a k information vector I then we would first multiply it by G to obtain the corresponding code word

$$V=IG \qquad 1.1.3.$$

Hence a linear code has 2^k code words or has dimension k over GF(2). The generator matrix G can be obtained from H [1].

The SEC Hamming codes are obviously easy to construct and can also be decoded in an easy fashion. A systematic technique for the construction of t-error correcting codes was not given until 1959, this is to say almost ten years after Hamming's paper. A class of codes which can correct all error patterns of weight t or less (or t or less random errors) was developed independently by Hocquenghem [9] in 1959 and by Bose and Chaudhuri [10] in 1960. This class is a subset of the class of cyclic codes. The class of cyclic codes is a subset of the class of linear codes.

DEFINITION 1.1.7.

If V is a linear code and if whenever $(a_1 .. a_n) \in V$, $(a_n\ a_1\ ..\ a_{n-1})$ also belongs to V then V is a cyclic code. $(a_n\ a_i .. a_{n-1})$ is the right cyclic shift of $(a_1\ ..a_n)$. This definition is the same for non-binary codes [1,2].

When speaking about cyclic codes it is more convenient to use polynomials instead of n-tuples. This is done by associating the polynomial $a_o+a_1x+....+a_{n-1}x^{n-1}$ with the n-tuple $(a_o, a_1, ...a_{n-1})$. It has been shown [1] that with every binary (n,k) cyclic code there is associated a unique polynomial G(x) of degree n-k such

that $G(x) \mid x^n - 1$. Every word in the code can be expressed as $C(x)\,G(x)$ where $C(x)$ is a polynomial of degree $k-1$ or less. The polynomial $G(x)$ is called the generator polynomial of the code. An important class of cyclic codes are the BCH (Bose-Chaudhuri-Hocquenghem) cyclic codes. The error correcting capability of the BCH cyclic codes is specified by the roots of the generator polynomial $G(x)$. The construction of such codes is given in Peterson [1]. The generator matrix of an (n,k) cyclic code generated by $G(x)$ is given by

$$G = \begin{bmatrix} G(x) \\ XG(x) \\ \vdots \\ X^{k-1}G(x) \end{bmatrix}$$

Moreover the parity check matrix is given as

$$H = \begin{bmatrix} (H(x))^* \\ \vdots \\ (X^{n-k-1}H(x))^* \end{bmatrix}$$

where $G(x)H(x) = x^n - 1$ and $(f(x))^* = x^n f(1/x)$ where n is the degree of $f(x)$. In 1961 W.W. Peterson [1, 13] provided a decoding procedure for the BCH codes. This break-through made the BCH codes very attractive from an implementation point of view. The BCH codes are very efficient for the correction of random errors. However in certain applications other type of errors predominate. One such type of error are the so

called burst errors which we define as

DEFINITION 1.1.8.

An error of the form,

$$(00\text{---}1a_{i_2}\ a_{i_3} \cdots a_{i_{L-1}}\ 100 - 0) \quad a_{i_K} = 0 \text{ or } 1,$$

is called a burst of length L starting in position i_1.

This type of error is common in telephone and recording systems. An efficient class of Burst-Error-Correcting cyclic codes are the Fire codes. These are also generated by a particular polynomial [18,1,2].

In the above discussion we started with linear codes. An even larger class of error-correcting codes are the group codes.

DEFINITION 1.1.9.

A code V whose elements form a group [16,17,5] under addition is called a group code.

In the case where the elements of V are binary n-tuples a group code is a linear code [1]. In this thesis since we will be mainly concerned with binary codes we will use the term group codes instead of linear codes.

We terminate this section with one last definition which will be used later.

DEFINITION 1.1.10.

Let V be a group code of length n. Let $GF(2)^n$ denote the group of all n-tuples under modulo 2 addition. Then V is a subgroup of $GF(2)^n$ and a coset of V in $GF(2)^n$ is defined for every $w \in GF(2)^n$ as

$$w+V = \{w+v \mid v \in V\}$$

CHAPTER 2

SYNCHRONIZATION

2.1. The Problem of Synchronization

When transmitting digital data over a channel, the receiver must be kept in synchronism with the transmitter. In other words, the receiver has to know when a message starts and when it ends. For example in an English text we encounter such symbols as , . ; : and so on. In a sense we could say that these symbols are used to maintain synchronism between the reader and the writer. If in an English text we were to remove such symbols then the new text could have many different meanings. To confuse the issue further we could remove the spaces between words.

To indicate the importance of proper synchronization between the receiver and the transmitter we note that in teletype communication for example, 53 out of every 153 msec of message space is alloted for synchronization purposes. It is also evident that Morse was aware of this problem since in the design of his code he allowed for times of open line between words and sentences. In most data channels, such as those linking computer users and a central computer in a time sharing system, as much as 30% of the channel capacity is being used for synchronization purposes.

In systems where coding is used for error control we have a similar problem. In this latter situation, when we wish to transmit k information symbols, we anex to these r redundancy symbols according to some predetermined rule, and then transmit the total k+r symbols. This type of coding is commonly known as block coding [1]. Inherent in block coding is the fact that an incoming sequence of digits is only meaningful when considered as a sequence of blocks of digits, each block corresponding to some transmitted word. It is perhaps worthwhile mentioning that not all coding schemes are of this form. For example in sequential coding [30] information symbols are coded individually or one by one instead of k by k. In this type of coding scheme the problem of synchronization is not so acute.

We now turn to a more specific formulation of synchronization. Suppose we have a code V of length n and transmit words from V over a channel. At the receiver, the incoming sequence has to be appropriately partitioned into blocks, each block corresponding to the transmitted word which caused it. Since all words have the same length, the problem of appropriate partitioning becomes one of finding the correct start of a code word in a stream of 2n digits.

If in a received sequence,

$$x_1 x_2 x_3 \ldots\ldots\ldots, \qquad 2.1.1.$$

we pick x_{i-L} or x_{i+R} as the start of a code word, where in reality it is x_i, then in either case a synch (synchronization) error has occured. In the first case we refer to the synch error as a loss of L bits or a slip to the left of L bits. The second case is called a gain of R bits or a slip to the right by R bits. Note that in 2.2.1 the first symbol arriving at the receiver is x_i. Both situations are shown in Figure 3.

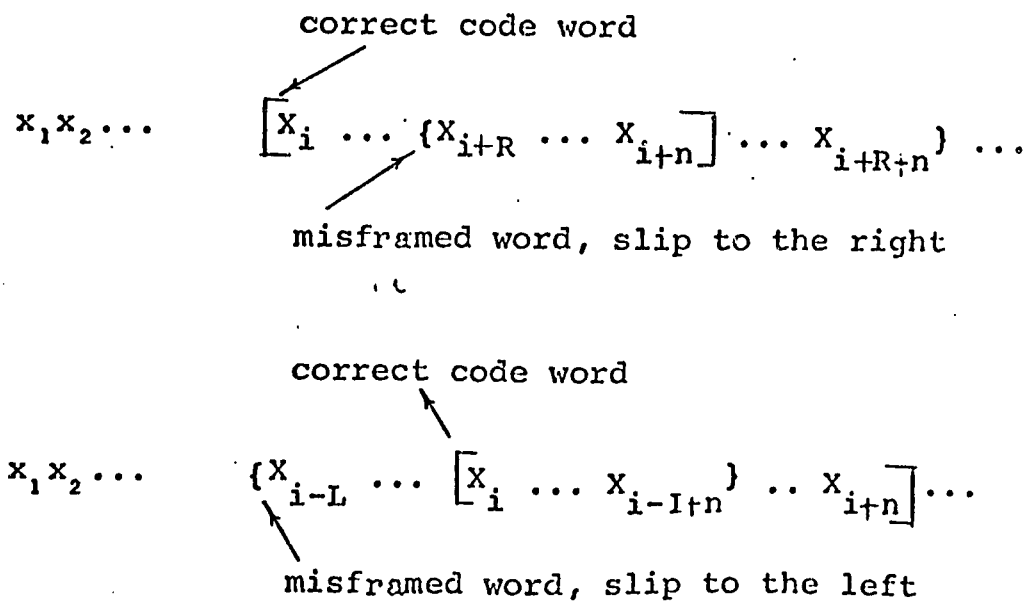


FIGURE 3

In the above description of slip it becomes apparent, after some thought, that we have to establish upper bounds on the value of L and R. This is clarified by the following consideration; let A,B, and C be any three code

words each of length n. Consider the sequence A,B,C which is,

$$a_1 a_2 \cdots a_n b_1 b_2 \cdots b_n c_1 c_2 \cdots c_n \ . \qquad 2.1.2.$$

If from 2.2.2. we choose the sequence,

$$a_{n-i+1} \cdots a_n b_1 b_2 \cdots b_{n-i} \ , \qquad 2.1.3.$$

then this can be interpreted in two ways. Expression 2.1.3. can either be considered as a left slip of i bits in B or a right slip of n-i bits in A. It is natural to say that if $i<n-i$, then the first case is more probable and if $n-i<i$, then the latter case is more probable. The next thing is, what do we do when $n-i = i$? This last situation will only occur when n is even. So we make the following restrictions:

1) If n is odd then $L,R \leq \frac{n-1}{2}$ 2.1.4.

2) If n is even then $L<n/2$, $R \leq n/2$. 2.1.5.

Having established the above bounds on L and R the problem of slip becomes well defined.

2.2. Mathematical Formulation of Slip

The power of analysis in science and engineering is self-

evident. One could say that mathematics is to the physical scientist what the microscope is to the biological scientist. An interesting thing in this context is that in many cases a physical problem can be formulated in more than one way. In such cases, the different formulations may reveal different aspects of the same problem and even point to different solutions. The criterion for judging the value of any one solution, at least in Engineering, should be its practical implications.

So far two main analytical techniques have been used to study the problem of synchronization. In one case the matrix representation of a code is used. This technique is applicable to all linear codes. The other technique uses the fact that a cyclic code can be represented by a polynomial G(x). This last method is more specific and consequently more simple. The matrix technique was developed by J. J. Stiffler in 1965 [21]. The polynomial technique was developed by S. Y. Tong in 1963[22] and Levy[39]. Along this line we should also mention Barker [20] who was the first person to investigate the problem of slip as such. In this thesis the polynomial formulation of slip will be used. Nevertheless, for the sake of completness, the matrix technique will be briefly described.

A. Matrix Technique

Let V be a linear code with generator matrix G and parity check matrix H. The elements of V form a vector space over $GF(2)=\{0,1\}$. The rows of G are linearly independent and span V, this is to say, the rows of G form a basis for V. G is a $k \times n$ matrix and H is an $(n-k) \times n$ matrix and $GH^T=0$. For the analysis of slip it is more convenient to use G^T, the transpose of G, instead of G. In this latter situation we consider the elements of V as $n \times 1$ vectors and any such vector V can be expresses as

$$V = G^T X \qquad 2.2.1.$$

where X is a $k \times 1$ column vector.

Let A, B and C be any three code words from V with;

$$A=G^T X_2 \ , \ B=G^T X, \ C=G^T X_1 \ .$$

Consider now the sequence $A^T B^T C^T$ which is

$$a_1 a_2 \ldots a_n b_1 b_2 \ \ldots \ b_n c_1 c_2 \ \ldots \ c_n. \qquad 2.2.2.$$

Let B^T be the word which we wish to decode from 2.2.2. If instead of B^T we select

$$a_{n-L+1} \ \ldots \ a_n b_1 b_2 \ \ldots \ b_{n-L} \qquad 2.2.3.$$

then a slip of L bits to the left has taken place. Similarly if instead of B^T we pick

$$b_{R+1} \cdots b_n c_1 c_2 \cdots c_R \qquad 2.2.4.$$

then a right slip of R bits has occured. In 2.2.3. and 2.3.4., L and R are bounded as discussed earlier.

Let us now partition G^T into G_1 and G_2 as

$$G^T = \begin{bmatrix} G_1 \\ \text{-----} \\ G_2 \end{bmatrix} \qquad 2.2.5.$$

where: a) G_1 is an (n-L)xK matrix for a left slip;and

b) G_1 is an (RxK) matrix for a right slip.

With this convention it follows that 2.2.3. and 2.2.4. can be expressed as:

$$(a_{n-L+1} \cdots a_n b_1 b_2 \cdots b_{n-L})^T = \begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_L \begin{bmatrix} \bar{X}_2 \\ X \end{bmatrix} \qquad 2.2.6.$$

and

$$(b_{R+1} \cdots b_n c_1 c_2 \cdots c_R)^T = \begin{bmatrix} G_2 & 0 \\ 0 & G_1 \end{bmatrix}_R \begin{bmatrix} \bar{X} \\ X_1 \end{bmatrix} \qquad 2.2.7.$$

where the subscript L refers to partitioning a) and the subscript R refers to partitioning b).

Expressions 2.2.6. and 2.2.7. are the mathematical formulations of left and right slip we seek. For an application of this technique to the study of slip in cyclic codes see Stiffler [21] and Tavares and Fukada [23].

B. Polynomial Technique

Let V be a code of length n. In this section we use the polynomial representation of a code word. Let A(x), B(x), and C(x) be any three code words from V and consider the sequence A(x), B(x), C(x), which is,

$$a_0 a_1 \ldots a_{n-1}\ b_0 b_1 \ldots b_{n-1}\ c_0 c_1 \ldots c_{n-1} \qquad 2.2.8.$$

Assume that B(x) is the word which we wish to decode from 2.3.8. If instead of B(x) we pick

$$a_{n-L}+a_{n-L+1}x+\ldots a_{n-1}x^{L-1}+b_0x^{L}+\ldots b_{n-L-1}x^{n-1} \qquad 2.2.9.$$

then a slip to the left of L bits has occured.

Expression 2.2.9. can be rewritten as:

$$x^L\left[b_0+b_1X+\ldots b_{n-L-1}X^{n-L-1}\right]+X^L\left[b_{n-L}X^{n-L}+\ldots b_{n-1}X^{n-1}\right]$$

$$+\left[a_{n-L}+\ldots a_{n-1}X^{L-1}\right]+X^L\left[b_{n-L}X^{n-L}+\ldots b_{n-1}X^{n-1}\right] \qquad 2.2.10.$$

This last expression becomes, upon combining the first

two terms,

$$X^L B(x) + \left[a_{n-L} \cdots a_{n-1} X^{L-1}\right] + X^n \left[b_{n-L} + \cdots + b_{n-1} X^{L-1}\right] \quad 2.2.11.$$

Now since all expressions are considered modulo X^n+1 and since X^n modulo X^n+1 is 1, we have

$$X^L B(x) + \left[(a_{n-L}+b_{n-L}) + (b_{n-L+1}+a_{n-L+1})X + \cdots (a_{n-1}+b_{n-1}) X^{L-1}\right] \quad 2.2.12.$$

For the purpose of analysis we replace the second term in 2.2.12. by a general polynomial $U_L(x)$ of degree L-1 or less. Hence we have

$$X^L B(x) + U_L(x) . \quad 2.2.13$$

Expression 2.2.13. is more general than 2.2.12. and hence results obtained using 2.2.13. will certainly be applicable to 2.2.12.

On the other hand, if from 2.2.8. we pick

$$b_R + b_{R+1} X + \cdots b_{n-1} X^{n-R-1} + C_0 X^{n-R} + \cdots C_{R-1} X^{n-1} \quad 2.2.14.$$

then a right slip of R bits has taken place.

Polynomial 2.2.14. can be rewritten as;

$$X^{-R}\left[b_R X^R + b_{R+1} X^{R+1} \cdots b_{n-1} X^{n-1}\right] + X^{-R}\left[b_0 + b_1 X + \cdots b_{R-1} X^{R-1}\right]$$

$$+ X^{n-R}[c_0+c_1X \;..\; c_{R-1}X^{R-1}] + \bar{X}^R[\, b_0+b_1X \;..\; b_{R-1}X^{R-1}] \qquad 2.2.15$$

Combining the first two expressions we have,

$$\bar{X}^R B(X)+\bar{X}^R[C_0\;\; C_1X \;..\; C_{R-1}X^{R-1}]+ \bar{X}^R[b_0\;\; b_1X \;..\; b_{R-1}X^{R-1}] \qquad 2.2.20$$

where X^n has been dropped for the same reason as in the case of a left slip. Finally we combine the last two terms in 2.2.20 to get

$$\bar{X}^R B(X) + \bar{X}^R[(b_0+C_0)+(b_1+C_1)X \;...\; (b_{R-1}+C_{R-1})X^{R-1}] \qquad 2.2.21$$

Once again we replace the last term in 2.2.21 by a general polynomial $U_R(X)$ of degree R-1 or less. This subtitution gives the desired form as:

$$\bar{X}^R\, B(X) + \bar{X}^R U_R(X). \qquad 2.2.22$$

Expressions 2.2.13 and 2.2.22 represent the situations of a left and a right slip respectively. These expressions do not allow for additive errors. The next thing we would like to do is to derive expressions analogous to 2.2.13 and 2.2.22 for the more general case when additive errors are present.

Let V be a code which can correct all error patterns belonging to a class E. In other words E is the set of all error patterns which can be corrected by V. For example if V is a random

e-error correcting code, then E is the set of all n-tuples of weight e or less. Suppose now we transmit three words A(X), B(X) and C(X) from V over a noisy channel which introduces errors from E. At the receiver the sequence will be

$$A^1(X),\ B^1(X),\ C^1(X), \qquad 2.2.23$$

where $A^1(X)=A(X)+E_1(X)$

$$B^1(X)=B(X)+E_2(X) \qquad 2.2.24$$

$$C^1(X)=C(X)+E_3(X)$$

and $E_i(X)\ \varepsilon E$ for i=1,2 and 3. Now once more we wish to select from 2.2.23 the word $B^1(X)$. At this point there is no use going through the same analysis as before if we make a few trivial observations. Let us first consider the case of a left slip. In expression 2.2.11, the only term coming from A(X) is $T_L(X)= a_{n-L}\ ..+a_{n-1}X^{L-1}$. In this case the only difference is that A(X) is corrupted by $E_1(X)$. The worst that could happen is that $E_1(X)$ be completely contained in $T_L(X)$, in which case $T_L(X)$ becomes $T_L(X)+E_1(X)$. In 2.2.23 B(X) becomes $B(X)+E_2(X)$. Assuming $E_1^1(X)$ is picked up with $T_L(X)$, 2.2.11 becomes,

$$X^L(B(X)+E_2(X)) + T_L(X)+E_1^1(X)+X^n[b_{n-L}\ ..\ b_{n-1}X^{L-1}] \qquad 2.2.25$$

If we now drop X^n and combine $T_L(X)$ with $b_{n-L} \cdots b_{n-1}X^{L-1}$ we have

$$X^L B(X) + U_L(X) + E_1^1(X) + X^L E_2(X) \qquad 2.3.26$$

Now since $U_L(X)$ is a general polynomial anyway we can incorporate $E_1^1(X)$ into it since $E_1^1(X)$ is at most of degree L-1. Hence we have

$$X^L B(X) + U_L(X) + E_2(X) \qquad 2.3.27$$

where X^L multiplying $E_2(X)$ is dropped.

The same arguments provide us with

$$\bar{X}^R B(X) + \text{-}RU_R(X) + E_2(X) \qquad 2.3.28$$

for a right slip of R bits in B(X) and where additive errors are present.

At this point we have a complete description of slip both in the presence and absence of additive errors. We collect these expressions in the form of a Table.

CASE	LEFT SLIP OF L BITS	RIGHT SLIP OF R BITS
NOISELESS	$X^L B(X) + U_L(X)$	$\bar{X}^R B(X) + \bar{X}^R U_R(X)$
NOISY	$X^L B(X) + U_L(X) + E(X)$	$\bar{X}^R B(X) + \bar{X}^R U_R(X) + E(X)$

TABLE 2.3.1.

In chapter 3 a short survey is made of the existing schemes to correct slip errors. In chapter 4 a new solution to the problem of synchronization of cyclic codes is presented. Lastly in chapter 5 some concluding remarks are made about the different schemes presented in the previous chapters.

N.B. Part of the results in chapter 4 have been presented elsewhere [38].

CHAPTER 3

SOME EXISTING TECHNIQUES

3.1. Brief Survey

In the past, a variety of methods for the minimization of incorrect decisions due to slip errors have been investigated. One of two drawbacks is almost invariably associated with these techniques. In some cases the technique is such that it makes poor use of the channel capacity. In other cases, the technique is only efficient in the absence of additive errors. An ideal technique then would be one which makes good use of the channel capacity and which is able to correct both slip and additive errors. In this work we are mainly concerned with the design of "synchronizable error correcting codes", that is to say, codes which can handle both slip and additive errors [26]. Since there already exists some classes of codes which are both easily implemented and decodable, then it would seem logical to direct our efforts to the modification of such codes in order to use them for the correction of slip in the presence of noise. Predominant among this class of codes are the BCH codes which are efficient at the correction of random errors. Unfortunately, as will be seen later, cyclic codes are highly vulnerable to synchronization errors.

A large proportion of the recent literature on Synchronization is about the modification of cyclic codes in order to make them synchronizable [22, 23, 26, 27]. Some work has also been done on the synchronization of Fire Codes [27]. We will now indicate a few of the techniques which have been developed so far. This survey is not at all complete and for a more exhaustive survey see Tavares [27] or Caldwell [28]. In the remaining discussion we will assume binary codes though many of the arguments are valid for non-binary codes.

If V is a binary code and we transmit words from V over a channel then the first attempt to synchronize this code would be to introduce a so called synchronizing symbol. This third symbol would then be used to separate code words. The introduction of a third symbol increases the complexity of the terminal equipment and hence from a practical point of view the method becomes less attractive. An even greater objection is that in the case where the channel is noisy, which is by far more realistic, the synch symbol may become indistinguishable from the binary symbols used in the code words. When this occurs incorrect decisions are bound to be made.

As an alternative to the above technique, a binary sequence P of length a is used instead of a new symbol [24]. During

transmission the prefix sequence P is placed at the beginning of each code word. Let A and B be any two code words, each of length n with

$$A=(a_1 a_2 \ldots a_n) \quad , \quad B=(b_1 b_2 \ldots b_n) \quad \text{and}$$

let $P=(p_1 p_2 \ldots p_\alpha)$. Then if we wish to transmit AB, we would instead transmit PA PB. This last sequence would be

$$p_1 p_2 \ldots p_\alpha a_1 a_2 \ldots a_n p_1 p_2 \ldots p_\alpha b_1 b_2 \ldots b_n \; . \qquad 3.1.1.$$

The sequence P is chosen in such a way that any sequence of α digits (except P itself) chosen from 3.1.1. will differ from P in at least one position. Once more if additive errors are admitted then some sequences other than P will in some cases be identical with P. In such cases a slip error will not be detected. The above two techniques of course are applicable to any binary code.

We now briefly discuss the concept of comma-free codes [21,29,23,27,22]. Let V be a binary group code with words of length n. Choose from V any three code words A,B and C and form the sequence,

$$a_1 a_2 \ldots a_n b_1 b_2 \ldots b_n c_1 c_2 \ldots c_n \; . \qquad 3.1.2.$$

Assume that the code word to be decoded from 3.1.2. is B. If instead of B we pick,

$$a_{n-i+1} \ldots a_n b_1 b_2 \ldots b_{n-i} \; , \qquad 3.1.3.$$

then a left slip of i bits has occured. If the channel is noiseless, and if it so happens that the n-tuple in 3.1.3. is not a code word, then we will detect the slip error. On the other hand, if 3.1.3. belongs to the code, and since the channel is noiseless, the syndrome of 3.1.3. will be zero. Hence we will incorrectly assume that 3.1.3. is the word which was transmitted. For example if V is a binary cyclic code and if A=B, then 3.1.3. becomes

$$b_{n-i+1} \cdots b_n b_1 b_2 \cdots b_{n-i} \quad , \qquad 3.1.4.$$

which is a code word. In fact 3.1.4. is nothing else but a cyclic shift of B. Hence cyclic codes are vulnerable to slip because of their cyclic nature. It will be seen later that the vulnerability of cyclic codes to slip errors is even more pronounced in the presence of noise.

If from 3.1.2. we had chosen,

$$b_{i+1} \cdots b_n c_1 c_2 \cdots c_i \qquad 3.1.5.$$

then a right slip of i bits has taken place. The same arguments as above hold for this case.

DEFINITION 3.1.1.

If V is a binary code and if sequences 3.1.3. and 3.1.5. are not code words of V for all i,

$0<i<r$ 3.1.6.

and for all code words A,B, and C, then V has comma-free freedom r. Again r is limited to n/2 as previously discussed.

It follows immediately from this definition that in the absence of additive errors a code V having comma-free freedom r will detect all slips not exceeding r.

DEFINITION 3.1.2.

A code V of length n having comma-free freedon n/2 is said to be comma-free.

A code as defined by DEFINITION 3.1.1. is also called a code with a prefix property of degree r.

J. J. Stiffler [21] was the first to discover the fact that certain cosets of a binary cyclic code V exhibit prefix properties. This idea was further investigated by Tong [22] in 1966 and later by Tavares [23,27] in 1968. This latter technique is referred to as the 'Coset Code Technique'. This will be discussed in further detail in section 3.2. Tong and Tavares are the first ones to provide coset codes which can correct both slip and additive errors simultaneously.

A class of synchronizable error correcting codes having a decoding procedure was provided by Bose and Caldwell [26] in 1967. Their technique is specifically tailored for

use with the BCH codes. This technique will be discussed in greater detail in the next sections.

We could at this point describe other methods of dealing with synchronization, however since our main interest lies in the synchronization of cyclic codes this will not be done.

We now turn to a description of the coset technique.

3.2. COSET TECHNIQUE

We subdivide this section into two parts. In the first part we will assume the Noiseless case where additive errors are not allowed. In the second we will relax this condition by allowing the presence of additive noise. First let us discuss the general philosophy behind the coset technique.

Let V be a binary code of length n and let $Q_1(X), Q_2(X)$ and $Q_3(X)$ be any three code words from V. Let us now transmit the sequence

$$Q_1(X),\ Q_2(X),\ Q_3(X)\ . \qquad 3.2.1.$$

Assume we wish to pick $Q_2(X)$. In general, due to slip noise we will, instead of $Q_2(X)$, pick

$$\text{or} \quad R_L(X) = X^{L} Q_2(X) + U_L(X) \qquad 3.2.2.$$

$$R_R(X) = X^{-R} Q_2(X) + X^{-R} U_R(X) \qquad 3.2.3.$$

according as the slip error is to the left or to the right respectively. If before transmission we were to add to each word a fix vector C(X) then 3.2.2. and 3.2.3. would become

$$R_L^1(X)=X^L[Q_2(X)+C(X)]+U_L(X) \qquad 3.2.4.$$

and $$R_R^1(X)=X^{-R}[Q_2(X)+C(X)]+X^R U_R(X) \qquad 3.2.5.$$

Now upon reception of an n-tuple we add once more C(X) to it before decoding. Obviously if there is no slip error then the second addition of C(X) will cancel the first C(X) added. On the other hand, if a slip error as occured then hopefully the process of adding C(X) before and after transmission will yield an error pattern. Mathematically we have after addition of C(X)

$$R_L^1(X)+C(X) = X^L Q_2(X) + C(X)[1+X^L]+U_L(X) \qquad 3.2.6.$$

and $$R_R^1(X)+C(X) = X^{-R} Q_2(X) + C(X)[1+X^{-R}]+X^{-R}U_R(X) \qquad 3.2.7.$$

If in 3.2.6. and 3.2.7. L=R=0 then as expected we have

$$R_L^1(X)+C(X)=Q_2(X)=R_R^1(X)+C(X) \ , \qquad 3.2.8.$$

or we have the transmitted word $Q_2(X)$.

The problem now is to design a polynomial C(X) such that expressions 3.2.6. and 3.2.7. are not code words for all values of L and R

$$1\leq L,R \leq s. \qquad 3.2.9.$$

If we have such a polynomial, then our technique will detect all slips not exceeding S.

NOISELESS CASE

The purpose of this section is to state, prove and discuss certain theorems on the cosets of binary cyclic codes. The purpose of this section is not to give all existing theorems on coset codes. For a good collection of such theorems see Tavares [27] . The few theorems given will hopefully indicate the type of proof which is encountered in this area.

Let V be a binary cyclic code of length n and with generator polynomial G(X). Let d be the minimum distance of V where

$$d=2t+1. \qquad 3.2.10.$$

Hence V is a t-random error correcting code. If we let E be the collection of all n-tuple of weight t or less, then a necessary and sufficient codition for V to correct all patterns from E is that the residue of all patterns belonging to E be distinct and non-zero. Furthermore a polynomial of degree n-1 or less belongs to V if and only if its residue modulo G(X) is zero. One codition on G(X) for V to be cyclic is that

$$X^n+1=G(X)\ h(X), \qquad 3.2.11.$$

or G(X) divides X^n+1[1] . The residue of a polynomial W(X) modulo G(X) will be denoted by {W(X)}.

Before proceeding to theorems on coset codes we will first

establish a few lemmas which will be used at a later time.

LEMMA 3.2.1.

Let $W_1(X)$, $W_2(X)$, ... $W_m(X)$ be a collection of m polynomials each of degree n-1 or less. Then

$$\{\sum_{i=1}^{m} W_i(X)\} = \sum_{i=i}^{m}\{W_i(X)\} \quad . \qquad 3.2.12.$$

PROOF: By the division algorithmn for polynomials, we know that the polynomial $W_i(X)$ can be uniquely written as

$$W_i(X) = Q_i(X)G(X) + r_i(X) \quad \deg r_i(X) \leq \deg G(X) \qquad 3.2.13.$$

where $Q_i(X)$ is called the quotient, $r_i(X)$ the remainder and deg means the"degree of" .

Hence from 3.2.13. we have that

$$\sum_{i=1}^{m} W_i(X) = G(X)\sum_{i=1}^{m} Q_i(X) + \sum_{i=1}^{m} r_i(X) . \qquad 3.2.14.$$

Now in 3.2.14. the degree of $\sum_{i=1}^{m} r_i(X)$ is less than deg $G(X)$ which means that $\sum_{i=1}^{m} r_i(X)$ is indeed the residue of $\sum_{i=1}^{m} W_i(X)$ modulo $G(X)$. This follows from the uniqueness of representation stated in the division Algorithmn. Equation 3.2.14. is equivalent to 3.2.12.

QED

LEMMA 3.2.2.

If V_Λ is an e random error correcting group code then an n-tuple W(X) does not belong to V if

$$0 < W(W(X)) < d=2e+1. \qquad 3.2.15.$$

PROOF: Since V is a group code, then the minimum distance of V is the minimum weight of V. By the minimum weight of V we mean $W(V(X))$ where V(X) is the non-zero word of V which has the least number of non-zero coeficients. This has to be so since the distance between (0,0,0, ... 0) and any code word W(X) is $W(W(X))$ and V is closed under addition. Therefore since the minimum distance is d then the minimum weight is d.

QED

LEMMA 3.2.3.

Let V be an (n,k) binary cyclic code generated by a polynomial G(X) of degree n-k. Then no burst of length n-k or less belongs to V.

PROOF: Because G(X) generates a cyclic code of length n, then

$$X^n+1 = G(X)\ H(X) \qquad 3.2.16.$$

or G(X) divides X^n+1.

Now let r(X) be a burst of length n-k or less. Starting in

in position j+1. Then

$$r(X) = X^j r_0(X) \qquad 3.2.17.$$

where $r_0(X)$ has degree less than n-k. Since G(X) divides X^n+1, X does not divide G(X), otherwise X would divide X^n+1 which is impossible. Therefore X^j and G(X) are relatively prime. If we assume that G(X) divides r(X) then

since G(X) and X are relatively prime G(X) must divide $r_0(X)$. Now since the degree of $r_0(X)$ is less than the degree of G(X) this last conclusion is impossible. Therefore r(X) does not belong to the code.

QED

The first theorem we state is due to Tavares [27].

THEOREM 3.2.1.

Given any (n, k) binary cyclic code, there exists a coset code which can determine both the magnitude and direction of any slip not exceeding (n-k-2)/2.

PROOF: Recall from Table 2.3.1. that in the noiseless case, the expressions for left and right slip are respectively:

$$R_L(X) = X^L W(X) + U_L(X) \qquad 3.2.18.$$

and

$$R_R(X) = X^{-R} W(X) + X^{-R} U_R(X). \qquad 3.2.19.$$

The words of the coset code are of the form

$W(X)+C(X)$ for $W(X)$ belonging to V and where $C(X)$ is the coset leader.

Transmitting words from the coset code and adding $C(X)$ to the received sequence we have

$$R_L^1(X)=X^L[W(X)+C(X)]+C(X)+U_L(X) \qquad 3.2.20.$$

and $$R_R^1(X)=X^{-R}[W(X)+C(X)]+C(X)+X^R U_R(X). \qquad 3.2.21.$$

In order for the coset code to do what is stated in the theorem it must

1) Distinguish between two left slips, say L_1 and L_2 with $L_1 \neq L_2$ and $1 \leq L_1, L_2 \leq S$,

2) Distinguish between two right slips, say R_1 and R_2 with $R_1 \neq R_2$ and $1 \leq R_1, R_2 \leq S$,

3) Distinguish between a left and a right slip, say of values L and R, $1 \leq L, R \leq S$.

Statements 1, 2, and 3 can be stated mathematically as:

1) $\{X^{L_1}W_1(X)+X^{L_1}C(X)+C(X)+U_{L_1}(X)\} \neq \{X^{L_2}W_2(X)+X^{L_2}C(X)+C(X) + U_{L_2}(X)\}$

for all $1 \leq L_1, L_2 \leq S$, $L_1 \neq L_2$ and any two code words $W_1(X)$ and $W_2(X)$. Using Lemma 3.2.1. and the fact that V is cyclic, the last expression is equivalently written as:

$$\{(X^{L_1}+X^{L_2})\,C(X)+U_{L_1}(X)+U_{L_2}(X)\} \neq \{0\} .$$

Finally assuming $L_2 > L_1$ we can incorporate $U_{L_1}(X)$ into $U_{L_2}(X)$ to obtain

$$\{(X^{L_1} + X^{L_2})\ C(X) + U_{L_2}(X)\} \not\equiv \{0\}. \qquad 3.2.22.$$

2) $\{X^{-R_1} W_1(X) + X^{-R_1} C(X) + C(X) + X^{-R_1} U_{R_1}(X)\} \not\equiv \{X^{-R_2} W_2(X) + X^{-R_2} C(X) + C(X) + X^{-R_2} U_{R_2}(X)\}$

for all $1 \leq R_i$, $R_2 \leq S$ $R_1 \neq R_2$ and for any two code words $W_1(X)$ and $W_2(X)$.

Once more this last expression can be simplified to

$$\{(1+X^{R_2-R_1})\ C(X) + U_{R_2}(X)\} \not\equiv \{0\} \qquad 3.2.23.$$

assuming $R_2 > R_1$.

3) $\{X^L W_1(X) + U_L(X) + X^L C(X) + C(X)\} \not\equiv \{X^{-R} W_2(X) + X^{-R} U_R(X) + X^{-R} C(X) + C(X)\}$

for all $1 \leq L, R \leq S$ and any two code words $W_1(X)$ and $W_2(X)$.

Reducing the expression we have:

$$\{(1+X^{L+R})\ C(X) + U_{L+R}(X)\} \not\equiv \{0\} \qquad 3.2.24.$$

where

$$U_{L+R}(X) = U_R(X) + X^{L+R} U_L(X) \qquad 3.2.25.$$

or a general polynomial of degree L+R-1 or less.

We now claim that if

$$C(X) = 1 + X^{n-1} \qquad 3.2.26.$$

then 1, 2, and 3 will be satisfied.

1) Upon substituting for C(X) the Left Hand Side (LHS) of 3.2.22. becomes

$$\{X^{L_1}+X^{L_2}+X^{L_1-1}+X^{L_2-1}\,U_{L_2}(X)\}$$

using the fact that $X^{n-1}=X^{-1}$.

Now since $L_1>L_2$ we can collect X^{L_1}, X^{L_1-1} and X^{L_2-1} with $U_{L_2}(X)$ to obtain

$$\{X^{L_2}+U_{L_2}(X)\} \quad . \qquad 3.2.27.$$

To make sure that 3.2.27. is not zero we use Lemma 3.2.2. First $W[X^{L_2}+U_{L_2}(X)]$ is never zero because X^{L_2} never overlapes with $U_{L_2}(X)$ and $L_2 \geq 1$.

Secondly we note that $X^{L_2}+U_{L_2}(X)$ is a burst of length at most S+1 when $L_2=S$. Then if we insist that

$$S+1 \leq n-k$$

$$\text{or} \quad S \leq n-k-1 \qquad 3.2.28.$$

then by Lemma 2.2.3. $X^{L_2}+U_{L_2}(X)$ is never a code word or 3.2.27 is satisfied.

2) If we analyse this case using the same arguments as in case 1 we obtain the same bound on S has given in 3.2.28. Therefore we omit its analysis and pass to case 3.

3) Substituting for C(X) in the L.H.S. of 3.2.24 we obtain:

$$\{1+X^{L+R}X^{n-1}+X^{L+R-1}+U_{L+R}(X)\} \quad .$$

Combining 1, X^{L+R-1} with $U_{L+R}(X)$ we have

$$\{X^{L+R}+X^{n-1}+U_{L+R}(X)\} \quad . \qquad 3.2.29.$$

Now $W(X^{L+R} + X^{n-1} + U_{L+R}(X))$ is never zero since X^{n-1} is never cancelled neither by X^{L+R} or $U_{L+R}(X)$. Secondly the expression $X^{n-1} + X^{L+R} + U_{L+R}(X)$ is a burst of length 2S+2 at most and if we insist that

$$2S+2 \leq n-k$$

or $$S \leq (n-k-2)/2 \qquad 3.2.30.$$

then by Lemma 3.2.3., $X^{n-1} + X^{L+R} + U_{L+R}(X)$ is never a code word or 2.2.29. is never zero.

Since $(n-k-2)/2 < n-k-i$ we have that

$$S=(n-k-2)/2$$

QED

It is worthwhile to collect, in a tabular form, the salient expressions developed in the above proof. These expressions are applicable to any coset code of a binary cyclic code with coset leader C(X).

Task to be performed by the Coset Code	condition to be satisfied by the Coset Leader C(X)
Differentiate between any two left slips L_1 and L_2, $L_2 > L_1$, $1 \leq L_1, L_2, \leq S$.	$\{(X^{L_1}+X^{L_2})\, C(X) + U_{L_2}(X)\} \neq \{0\}$
Differentiate between any two right slips R_1 and R_2, $R_2 > R_1$, $1 \leq R_1 R_2 \leq S$.	$\{(1+X^{R_2-R_1})C(X)+U_{R_2}(X)\} \neq \{0\}$
Differentiate between a left and a right slip of magnitude L and R, $1 \leq L, R \leq S$	$\{(1+X^{L+R})\, C(X)+ U_{L+R}\} \neq \{0\}$

TABLE 2.2.1.

In Theorem 3.2.1. we are essentially using the additive error correcting capability of the code V to correct slip errors. In the more general case when we wish to correct both additive and slip errors we use part of the additive error correcting capability of the original code to correct slip errors. Another alternative is to make a trade-off between the transmission rate of V and the slip error correcting capability. We define the transmission rate of an (n,k) code as the ratio k/n. So what we are essentially saying is that we could conceptually start with a code V having transmission rate R, construct from V a new code V^1 (essentially a subcode) having transmission rate $R^1<R$ such that V^1 has the same additive error correcting capability as V but can also correct slip errors (for example the technique of Section 3.3.). In any case if we wish to use an error correcting code V to correct also slip errors then a price will have to be paid somewhere*; either a lowering of the additive error correcting capability of V or a lowering of the transmission rate of V or both. Whichever is used will depend on the application in mind.

Another point worth discussing is the ease of decodability of a code. From an implementation point of view it is always desirable to have a code which has an easy decoding algorithmn. Though not universal, a code having an easy decoding algorithmn usually has a lower transmission rate, everything else being equal. The ideal case of course is to have a code with as high a transmission rate as possible

* Unless some cosets are not used for the correction of additive errors.

and as easy a decoding procedure as possible.

As mentioned earlier, the coset code obtained in Theorem 3.2.1. can determine both the magnitude and direction of any slip error not exceeding (n-k-2)/2. This means that when such a code is used in a practical system the decoder or corrector can correct the slip error in the next frame following the one in which the slip is detected. This of course is not the only way to correct slip errors. We could for example use a code which can only determine the direction or sign of the slip error. In this case the decoder would have to go through a search process before correct synchronization is regained.

A plausible search process or algorithmn using such a code could be the following:

STEP 1. Let B=S=maximum slip expected

STEP 2. Pick an n-tuple from the incoming sequence and compute the direction of slip.

STEP 3. If it is left go to STEP 4, if it is right go to STEP 5, if it is zero go to STEP 6.

STEP 4. Let B=B-1, shift to the right by B places and go to STEP 2.

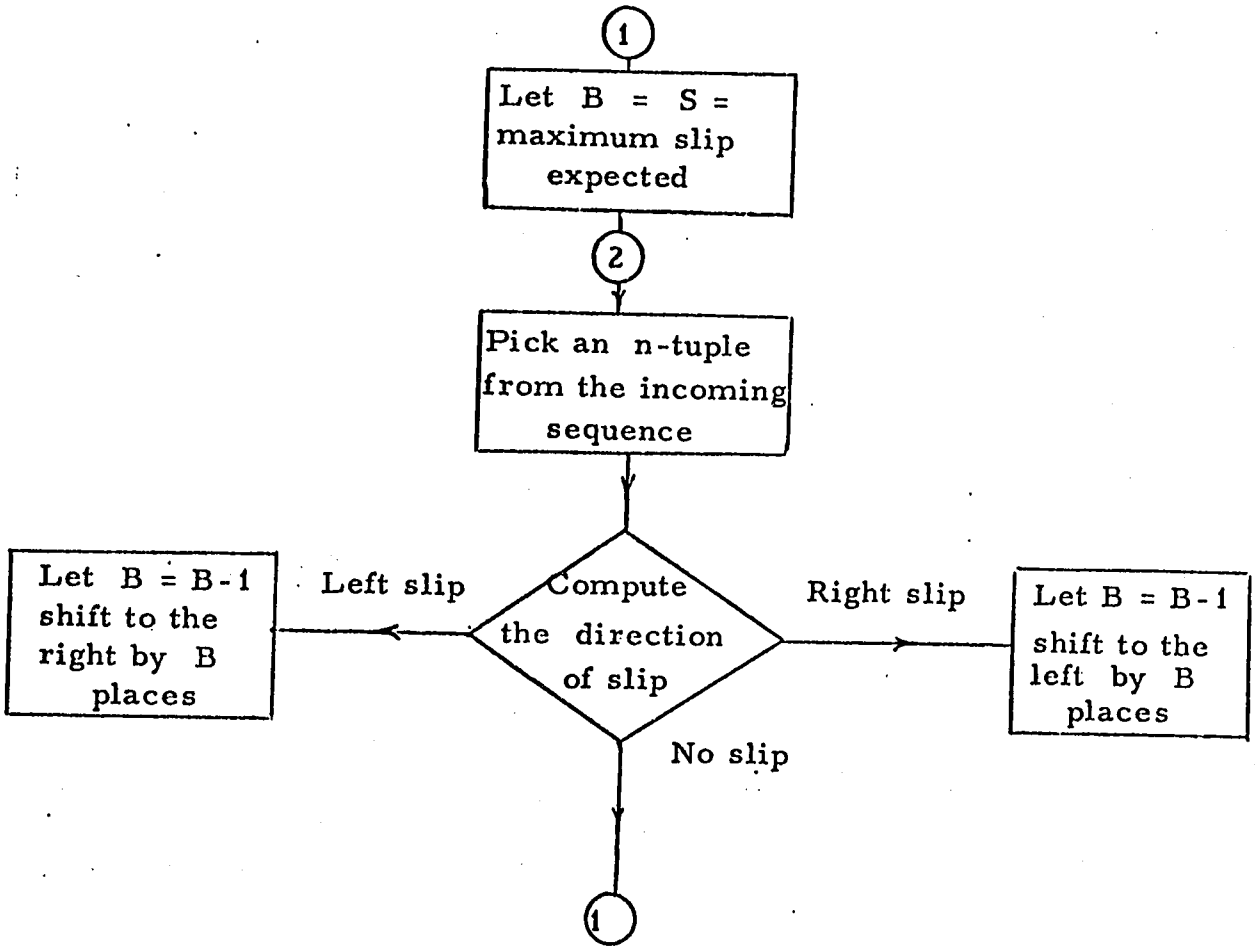
STEP 5. Let B=B-1, shift to the left by B places and go to STEP 2.

STEP 6. Correct synchronization is obtained, go to STEP 1.

With this system a certain number of slip errors have to be tolerated before correct synch is regained. On the other hand it places less of a demand on the capability of V. When

Flow Chart For Decoding Algorithm

Described on Page 40



only the direction of slip is required the bound on S can be raised to (n-k-1)/2.

An alternative to this last technique is to go through the search process using the word in which the slip is detected, the previous and the following word. Using this approach we do not let any errors go by uncorrected. This system however requires a large memory to store the incoming sequence while the decoder is finding the slip error somewhere up the line.

It is immediate from the above remarks that a code which can determine both the magnitude and direction of slip is more desirable from an implementation point of view. For a more detailed discussion of these ideas see Tavares [27].

Our next theorem is also due to Tavares and Fukada*. We do not give the proof.

THEOREM 3.2.2.

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

NOISY CASE

In this section we limit ourselves to one theorem. This theorem, in its final version, is due to Tavares and Fukada†.

* Theorem 2.5. in reference [27].

THEOREM 3.2.3.

Given an (n,k) binary cyclic code with minimum distance d, there exists a coset code which can correct both e or less additive errors and S or less slip errors even when they occur simultaneously in each received n-tuple. Furthermore S is given by

$$S=\text{MIN}\left[(d-4e-3)/2,\quad (n-e-2)/2(e+1)\right] . \qquad 3.2.31.$$

We will not prove this theorem in its details. For a proof see Tavares [27]. The coset leader which will satisfy the conditions stated in the theorem is

$$C(X)=X^{n-1}+\sum_{i=0}^{e} X^{i(2S+1)} \qquad 3.2.32.$$

The coset code will:

1. differentiate between a left slip and an additive error,
2. distinguish between a right slip and an additive error,
3. distinguish between two left slips L_1 and L_2 where $1\leq L_1, L_2 \leq S$ and $L_1 \neq L_2$,
4. distinguish between two right slips R_1 and R_2 where $1\leq R_1, R_2 \leq S$ and $R_1 \neq R_2$,
5. distinguish between a right and a left slip.

A tabulation of results using Theorem 3.2.3. is given in Table 3.2.1.

1	2	3	4	5
				Theo3.2.3.
(n,k,t)	d	k/n	e	S
(127,92,5)	11	.725	1 2	2 0
(127,85,6)	13	.670	1 2 3	3 1 0
(127,64,10)	21	.505	1 2 3 4 5	7 5 3 1 0
(127,22,23)	47	.173	1 2 3 4 5 6 7 8 9	20 18 15 12 10 8 7 6 4
(255,191,8)	17	.750	1 2 3 4	5 3 1 0
(255,163,12)	25	.690	1 2 3 4 5 6	9 7 5 3 1 0

TABLE 3.2.1.

LEGEND TO TABLE 3.2.1.

Column 1. (n,k,t) of the parent code V.

Column 2. d=2t+1 is the minimum distance of V.

Column 3. k/n is the transmission rate of V.

Column 4. e is the additive error correcting capability of the coset code according to Theorem 3.2.3.

Column 5. S is the maximum slip error correcting capability given by relation 3.2.31.

3.3. Bose Caldwell Technique

This section is an account of Bose and Caldwell's paper entitled "Synchronizable-Error-Correcting-Codes", see reference [26].

Let V be a (n,k) BCH code (as described in chapter 1) generated by G(X) and with minimum distance d. Then G(X) divides $X^{n}-1$ or

$$X^{n}-1 = G(X)\ H(X). \qquad 3.3.1.$$

Let γ be a root of H(X). γ not a root of G(X), with minimum polynomial m(X). m'(X) is the monic irreducible polynomial of least degree which has γ as a root. The coefficients of m'(X) are in GF(q) and $n=q^{m}$. For a discussion of these concepts see Peterson [1] chapter VI. The degree of m'(X) will be m_1 where m_1 divides m†. Also since γ is a root

† Theorem 11 page 128 of reference [17].

of $H(X)$ then $m(X)$ divides $H(X)$ [1]. Lastly let the order of γ be n_1 where n_1 divides n.

Let V^* be the subcode of V generated by $m(X)\ G(X)$. V^* is the cyclic code made up of all polynomials of V which are divisible by $m(X)\ G(X)$. It is an (n,k^*) cyclic code where $k^*=k-m_1$. Any word $W^*(X)=(\omega_0\ \omega_1\ ..\omega_{n-1})$ of V^* satisfies the equation

$$\omega^*(X)\ [H^T,\ H_1^T]=0 \qquad 3.3.2.$$

where H is the parity check matrix of V and

$$H_1=[1, \gamma,\ \gamma^2 \ldots \gamma^{n-1}]\ . \qquad 3.3.3.$$

Our objective is to construct a code which can correct S_L left slips or S_R right slips and the simultaneous occurence of e additive errors where

$$S^*=S_L+S_R<\ n_1>\ 1. \qquad 3.3.4.$$

Encoding Procedure

Let $C(X)=(C_0\ C_1\ \ldots\ C_{n-1})$ be a fixed non-null (or non-zero) word of V which does not belong to V^*. If we wish to transmit a word $\omega^*(X)=(\omega_0\ \ldots\ \omega_{n-1})$ from V^* we would transmit instead

$$t_a\ (X)=\omega_a^*(X)+C_a(X) \qquad 3.3.5.$$

where $$\omega_a^*(X)=\omega_{n-S_L}\ \omega_{n-S_L+1}\cdots\omega_{n-1}\ \ \omega_0\omega_1\cdots\omega_{S_R-1} \qquad 3.3.6.$$

and $$C_a(X)=C_{n-S_L}\ C_{n-S_L+1}\ \ldots\ C_{n-1}\ C_0\ C_1\ldots C_{n-1}\ C_0C_1\ldots$$

$$C_{S_R-1} \qquad 3.3.7.$$

Hence we are transmitting words from a code T which as q^{k_a} words of length $n_a=n+S_L+S_R$ and $K_a=K^*=K-m_1$. T is an (n_a,k_a) code.

DECODING PROCEDUPE

Let the additive error per code word be

$$e_a(X)=f_{n-S_L}\cdots f_{n-1}e_0e_1\cdots e_{n-1}f_0f_1\cdots f_{S_R-1}. \qquad 3.3.8.$$

STEP 1. Pick from the incoming sequence an n_a tuple and drop the first S_L and last S_R symbols. The truncated word has length n.

Case (i) If there are no slip errors then the truncated word will be

$$Y(X)=(\omega_0\omega_1..\omega_{n-1})+(C_0C_1\ \ldots\ C_{n-1})+(e_0e_1..\ e_{n-1}) \qquad 3.3.9.$$

$$=\omega_a(X)+C(X)+e'(X). \qquad 3.3.10.$$

Case (ii) If there is a left slip of $L\leq S_L$ places then the truncated word will be

$$Y(X)=(\omega_{n-L}\ \omega_{n-L+1}\cdots\omega_{n-1}\ \omega_0\omega_1\ \ \omega_{n-L-1})$$
$$+(C_{n-L}\ C_{n-L+1}\cdots C_{n-1}C\ C\ \ldots\ C_{n-L-1})$$
$$+(f_{n-L},\cdots f_{n-1},\ e_0e_1\ \ldots\ e_{n-L-1})$$
$$Y(X)=X^L\,W^*(X)+X^LC(X)+e''(X) \qquad 3.3.11.$$

Case (iii) If there is a right slip of $R \leq S_R$ places then the truncated word will be

$$\begin{aligned} Y(X) = &(\omega_R \cdots \omega_{n-1}\ \omega_0\omega_1 \cdots \omega_{R-1}) \\ &+ (C_R \cdots C_{n-1}\ C_0C_1 \cdots C_{R-1}) \\ &+ (e_R \cdots e_{n-1}\ \int_0 \int_1 \cdots \int_{R-1} \end{aligned}$$

$$Y(X) = X^{-R} W^*(X) + X^{-R} C(X) + e'''(X) \qquad 3.3.12.$$

STEP II We form the additive error syndrome

$$Y(X)\ H^T = e'(X) H^T \quad \text{in case (i)}, \qquad 3.3.13.$$

$$Y(X)\ H^T = e''(X) H^T \quad \text{in case (ii)}, \qquad 3.3.14.$$

$$Y(X)\ H^T = e'''(X) H^T \quad \text{in case (iii)}, \qquad 3.3.15.$$

If we assume that

$$W[e_a(X)] \leq (d-1)/2 \qquad 3.3.16.$$

then all additive errors will be correctable. In other words $e'(X)$, $e''(X)$ and $e'''(X)$ are all correctable by V.

STEP III The received truncated word $Y(X)$ is now corrected. We thus obtain

$$Z(X) = \omega^*(X) + C(X) \quad \text{in case (i)} \qquad 3.3.17.$$

$$Z(X) = X^L \omega^*(X) + X^L C(X) \quad \text{in case (ii)} \qquad 3.3.18.$$

$$Z(X) = X^{-R} \omega^*(X) + X^{-R} C(X) \quad \text{in case (iii)} \qquad 3.3.19.$$

STEP IV We now form the slip-error syndrome

$$Z(X)\ H_1^T. \qquad 3.3.20.$$

This computation is

$$Z(X)H_1^T = C(X)H_1^T \quad \text{in case (i),} \qquad 3.3.21.$$

$$Z(X)H_1^T = X^L C(X)H_1^T \quad \text{in case (ii),} \qquad 3.3.22.$$

$$Z(X)H_1^T = X^{-R} C(X)H_1^T \quad \text{in case (iii),} \qquad 3.3.23.$$

Since V* is also cyclic by our method of construction.

Recalling from 3.3.3. that

$$H_1 = \left[1\ \gamma\ \gamma^2\ \ldots\ \gamma^{n-1}\right]$$

we have

$$C(X)H_1^T = C_0 + C_1\gamma + C_2\gamma^2 + \ldots C_{n-1}\gamma^{n-1} = \delta\ , \qquad 3.3.24.$$

where δ is an element of $GF(q^m)$ which we know since we know C(X) and H_1.

Again,

$$X^L C(X)H_1^T = \gamma^L \delta \qquad 3.3.25.$$

and

$$X^{-R} C(X)H_1^T = \gamma^{n-R}\delta = \gamma^{n_1-R}\delta \qquad 3.3.26.$$

Since $\gamma^n = \gamma^{n_1}$.

Step $\overline{V}$ Divide the slip-error syndrome obtained in $\overline{IV}$ by the known element δ to obtain

$$\begin{cases} 1 & \text{in case (i),} \\ \gamma^L & \text{in case (ii),} \\ \gamma^{n_1-R} & \text{in case (iii),} \end{cases}$$

Now since $S_L + S_R < n_1$ then $n_1 - S_R > S_L$ or $n_1 - R > S_L$. Hence if the answer to step $\overline{V}$ is 1, we conclude that no slip error has occured. Secondly if the answer is γ^μ and $\mu > S_L$ then we conclude that a right slip of magnitude $n_1 - \mu$ has occurred. Lastly if the answer is γ^μ and $\mu < S_L$ then a left slip of magnitude μ has occurred.

By applying a reverse shift to Z(X) and adding C(X) to it we obtain the code word w(x) from V*.

Therefore V* is a $((d-1/2), S^*)$ synchronizable error correcting code.

EXAMPLE:

Let V be the (15,7) BCH code generated by

$$G(X)=1+X^4+X^6+X^7+X^8.$$

Now $15=2^4-1$ and the roots of G(X) belong to the Galois Field $GF(2^4)$ generated by $1+X+X^4$. For a tabulation of the elements of $GF(2^4)$ see Peterson [1]. The roots of G(X) are:

$$\alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^6, \alpha^8, \alpha^9, \alpha^{12}$$

where α is a primitive root of $1+X+X^4$. V is a 2 additive error correcting code.

Let us assume we wished to correct 1 left and 1 right slip error or $S^*=1+1=2$. Hence we must choose an element from $GF(2^4)$ whose order n_1 is larger than 2. The elements α^5 and α^7 both satisfy this condition. In order to minimize the redundancy we choose that element which has a minimal function of least degree. In this case we choose $\gamma=\alpha^5$ whose minimal function is

$$f(X)=1+X+X^2.$$

Hence the subcode V* is generated by

$$G^*(X)=G(X)\ f(X) = 1+X+X^2+X^4+X^5+X^8+X^{10}.$$

V* is a (15, 5) code with minimum distance 7.

We now compute H and H_1.

$$H=\begin{bmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \cdots\cdots & \alpha^{14} \\ 1 & \alpha^3 & (\alpha^3)^2 & & & & \cdots\cdots & (\alpha^3)^{14} \end{bmatrix}$$

$$H= \begin{bmatrix} 1&0&0&0&1&0&0&1&1&0&1&0&1&1&1 \\ 0&1&0&0&1&1&0&1&0&1&1&1&1&0&0 \\ 0&0&1&0&0&1&1&0&1&0&1&1&1&1&0 \\ 0&0&0&1&0&0&1&1&0&1&0&1&1&1&1 \\ 1&0&0&0&1&1&0&0&0&1&1&0&0&0&1 \\ 0&0&0&1&1&0&0&0&1&1&0&0&0&1&1 \\ 0&0&1&0&1&0&0&1&0&1&0&0&1&0&1 \\ 0&1&1&1&1&0&1&1&1&1&0&1&1&1&1 \end{bmatrix}$$

and

$$H_1= \begin{bmatrix} 1 & \gamma & \gamma^2 & \cdots\cdots & \gamma^{14} \end{bmatrix}$$

$$H_1= \begin{bmatrix} 1&0&1&1&0&1&1&0&1&1&0&1&1&0&1 \\ 0&1&1&0&1&1&0&1&1&0&1&1&0&1&1 \\ 0&1&1&0&1&1&0&1&1&0&1&1&0&1&1 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \end{bmatrix}$$

We pick C(X)= (1 0 0 0 1 0 1 1 1 0 0 0 0 0 0) and let us choose from V* the word G*(X)$(1+X+X^3)$ or $w_a^*(X)$=(1 0 0 0 0 1 1 1 0 1 1 0 0 1 0)

Now $C_a(X)$ = (0 1 0 0 0 1 0 1 1 1 0 0 0 0 0 0 1)

and $w_a^*(X)$= (0 1 0 0 0 0 1 1 1 0 1 1 0 0 1 0 1).

The transmitted word is

$$t_a(x) = w_a^*(x)+c_a(x)=(0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0).$$

Let $e_a(x)$ = (1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0), therefore the corrupted vector is (1 0 0 0 0 1 0 0 0 1 1 1 0 0 1 0 0).

Step 1. Assume a slip to the left by 1 bit and assume that the bit preceeding the word is 1. Hence the truncated word is

Y(X)= (1 0 0 0 0 1 0 0 0 1 1 1 0 0 1).

Step II Compute $Y(X)\,H^T$ and find the corresponding additive error $e_a(x)$.

Step III The received truncated vector after correction is

$$Z(X) = (0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1).$$

Step IV Compute the slip error syndrome

$$Z(X)\,H^T$$

which is $(1\ 0\ 0\ 0) = \alpha^{15}$

Step V Divide the slip syndrome by $c(\gamma) = \delta$.

$$c(\gamma) = \delta = X^{10}.$$

Therefore $\alpha^{15}/\alpha^{10} = \alpha^5 = \gamma^1$. The order of α^5 is $3 = n_1$. Now $\mu = 1$ $S_L = 1$ and hence we conclude that a left slip of magnitude $\mu = 1$ has occurred, which is correct.

Finally

$$w_a(x) = X^{-1}Z(X) + c(x) = (1\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0).$$

END OF EXAMPLE

The transmission rate of V* is

$$k_a/n_a = (k - m_1/n + S^*)\quad k/n \qquad 3.3.27.$$

Note that the Bose-Caldwell technique does not alter the additive error correcting capability of the code. On the other hand the transmission rate of V* is less than that of V as given in 3.3.27. This illustrates the concept discussed in section 3.2.

We now give a Table of results using the Bose-Caldwell technique.

1	2	3	4	5	6	7	8	9	10
n, k, t	G(X)	k/n	γ	n_1	m_1	f(x)	G*(x)	n_a, k_a, t, S^*	k_a/n_a
(15, 11, 1)	23	.734	α^3	5	4	37	721	(18, 7, 1, 3)	.389
			α^5	3	2	7	171	(17, 9, 1, 2)	.530
(15, 7, 2)	721	.467	α^5	3	2	7	2067	(17, 5, 2, 2)	.294
(31, 21, 2)	3551	.627	α^5	31	5	67	107657	(61, 16, 2, 30)	.262
								(47, 16, 2, 16)	.340
								(39, 16, 2, 8)	.410
								(35, 16, 2, 4)	.458
(31, 16, 3)	107, 657	.517	α^7	31	5	57	5, 474, 225	(61, 11, 3, 20)	.180
								(47, 11, 3, 16)	.234
								(33, 11, 3, 2)	.333
(63, 51, 2)	12, 471	.810	α^{21}	3	2	7	65, 657	(65, 49, 2, 2)	.755
			α^9	7	3	15	161, 025	(69, 48, 2, 6)	.696
								(67, 48, 2, 4)	.717
(63, 45, 3)	1, 501, 317	.715	α^{21}	127	7	345		(253, 85, 5, 126)	.662
	1, 467							(147, 85, 5, 20)	.580
								(137, 85, 5, 10)	.620
								(133, 85, 5, 6)	.640
(127, 85, 6)		.670	α^{21}	127	7	345		(133, 78, 6, 6)	.586
(127, 64, 10)		.505	α^{21}	127	7	345		(137, 57, 10, 10)	.416
(127, 22, 23)		.173	α^{47}	127	7	271		(167, 15, 23, 40)	.090
								(147, 15, 23, 20)	.102
								(135, 15, 23, 8)	.111
								(129, 15, 23, 2)	.116
(255, 191, 8)		.750	α^{17}	15	4	43		(269, 187, 8, 14)	.695
								(265, 187, 8, 10)	.706
								(257, 187, 8, 2)	.727

TABLE 2.2.1

LEGEND TO TABLE 3.3.1.

Column 1. (n, k, t) of the parent code V t is the additive error correcting capability of V.

Column 2. G(X) is the generator polynomial of V. If $G(X) = \sum_{i=0}^{n} c_i X^i$ then the entry is $\sum_{i=o}^{n} C_i \, 2^i$ base 8.

Column 3. k/n is the transmission rate of V.

Column 4. γ is a root of H(X) where $X^n - 1 = G(X)H(X)$, γ not a root of G(X).

Column 5. n_1 is the order of γ. Recal that $S^* \leq n_1$.

Column 6. m_1 is the degree of the minimal function of γ.

Column 7. f(x) is the minimal function of γ.

Column 8. G*(x) is the generator polynomial of V*. It is given up to the (127, 92, 5) code.

Column 9. (n_a, k_a, t, S^*) of V*: $n_a = n + S^*$, $k_a = k - m_1$, $S^* = S_L + S_R$

Column 10. k_a/n_a is the transmission rate of V*.

CHAPTER 4

4.1. Introduction

We have seen in Chapter 3 two techniques which enable cyclic codes to correct both additive and slip errors. The coset technique uses some of the additive error correcting capability of the parent code V to correct slip errors. In other words if the parent code V is a t-random error correcting code then there exists a coset code which can correct $e<t$ random errors and s slip errors. A decoding algorithmn is not provided for the coset codes. Nevertheless the results on coset codes are important for at least two reasons.

1. It is possible that later on a decoding algorithmn will be found for these codes.

2. They provide a yardstick to evaluate the performance of other synchronizable error correcting codes.

The results on coset codes in Chapter 3 were derived for binary cyclic codes. There exist however a variety of results on the cosets of non-binary cyclic codes [27].

The Bose-Caldwell technique derived in Chapter 3 section 3 is specifically tailored for BCH codes*. It is applicable to binary as well as non-binary BCH codes. More importantly it has a relatively simple decoding algorithmn. The Bose and Caldwell technique does not alter the additive error correcting capability of the parent code V. The added feature of slip

* The technique has been generalized by Weldon [35].

error correcting capability is counterbalanced by a degradation in the transmission rate.

In this Chapter we will construct a class of synchronizable error correcting codes which are both easily encoded and which have an extremely easy decoding procedure. The synchronizable error-correcting code obtained will have a lower transmission rate and a lower additive error correcting capability than the parent code. This latter technique is applicable to any class of cyclic codes having certain properties which will be described in the next section.

In section 4.2. we will derive a method of constructing a class of synchronizable error-correcting codes. Also an example will be given as to how the technique is applied. In section 4.3. a modified version of the technique described in section 4.2. will be described along with an example. Lastly a table of results similar to table 3.3.1. will be given.

4.2. Noiseless Case

As a motivating exercise let us devise a method of correcting slip errors in the absence of noise. Our first attempt to solve this problem will lead to a very inefficient but enlightening solution.

Let us first recall the expressions for slip errors in the absence of noise. From table 2.3.1. they are

$$X^{L}B(X)+U_{L}(X) \qquad 4.2.1.$$

for a left slip of L places and

$$X^{-R}B(X)+X^{-R}U_R(X) \qquad 4.2.2.$$

for a right slip of R places.

Furthermore,

$$U_L(X)=(a_{n-L}+b_{n-L})+(a_{n-L+1}+b_{n-L+1})X + \ldots (a_{n-1}+b_{n-1})X^{L-1} \qquad 4.2.3.$$

and $$U_R(X)=(b_0+c_0)+(b_1+c_1)X + \ldots (b_{R-1}+C_{R-1})X^{R-1} \qquad 4.2.4.$$

where $A(X)=a_0+a_1X+ \ldots a_{n-1}X^{n-1}$

$B(X)=b_0+b_1X+ \ldots b_{n-1}X^{n-1}$

$C(X)=c_0+c_1X+ \ldots c_{n-1}X^{n-1}$

and the sequence is A(X), B(X), C(X).

Now if we were to make

$$U_L(X)=U_R(X)=0 \quad \text{for all } 1\leq L,\ R\leq S^1$$

then expressions 4.2.1. and 4.2.2. would become

$$X^{L}B(X) \qquad 4.2.5.$$

and $$X^{-R}B(X) \qquad 4.2.6.$$

respectively. Our next problem would be to extract the value of L from 4.2.5. and the value of R from 4.2.6. One possibility is to choose only those B(X) from the parent code V such that the first coefficient in 4.2.5. has power L and such that the first coefficient in 4.2.6. has power -R. In other words we would choose a special subset of the parent code. We formally state the complete result in the form of

a Theorem .

THEOREM 4.2.1.

Let V be an (n,k) binary cyclic code. Then for every integer $S \leq (k-1)/2$ there exists a subset H of V which can correct all slip errors not exceeding S.

Proof: Let H be the subset of V consisting of all code words which start with exactly S 0's and which end with at least S 0's. For example a typical word of H would be

$$000 \ldots 01A_{S+1}A_{S+2} \quad A_{n-S-1}00 \ldots 0. \qquad 4.2.7.$$

It is immediate that if we transmit words from H then $U_L(X)$ and $U_R(X)$ will be identically zero.

Furthermore since all words of H have a 1 in the S+1 position then in 4.2.5. and 4.2.6. the first non-zero coefficient will have power S+L or S-R respectively.

The decoding procedure would be as follows:

STEP 1. Pick an n-tuple from the incoming sequence

STEP 2. Locate the first non-zero coefficient in the received n-tuple and call its power α.

STEP 3. If $\alpha < S$ then a right slip of magnitude $S-\alpha$ has taken place. If $\alpha > S$ then a left slip of magnitude $\alpha - S$ has taken place. Finally if $\alpha = S$ then no slip has taken place.

Since $1 \leq L, R \leq S$ the above algorithmn will always work.

Since the shortest code word in V has length n-k+1 (Lemma 3.2.3.) then the maximum value of S is $(n-(n-k+1))/2$. The number of words in H is 2^{k-2S-1} or the transmission rate of H is

$$\frac{k-2S-1}{n} = \frac{k}{n} - \frac{2S+1}{n} \qquad 4.2.8.$$

QED

The above procedure is inefficient since it does not make use of the error correcting capability of the parent code V. On the other hand the decoding procedure is extremely simple.

Our next step in attempting to solve the problem of slip is to use the additive error correcting capability of the parent code V.

THEOREM 4.2.2.

Let V be an (n,k) binary cyclic code which can correct any burst of length S^1 or less occuring in the first $2S^1$ places and another burst of length S^1 or less always occuring in the last $2S^1$ places. Let V^1 be the shortened (n^1,k^1) version of V and consisting of only those words which start with a 1, where $n^1=n-2S^1$ and $k^1=k-2S^1-1$. Then if we use V^1 for transmission and V for decoding, and provided a certain decoding Algorithmn is followed, V^1 can correct all slip errors not exceeding S^1. As before S^1 is upper bounded by $(k-1)/2$.

Proof: Since the proof is tied up with the decoding algorithmn we present it first.

DECODING PROCEDURE

Step 1. Pick an n^1 tuple Y(X) from the incoming sequence

Step 2. Multiply Y(X) by X^{S^1}. This is equivalent to shifting to the right by S^1 in a shift register of length n. The length being n since we use V for decoding.

Step 3. Decode $X^{S^1}Y(X)$ using the existing decoding procedure for V. Call the corrected vector R(X).

Step 4. Locate the first non-zero coefficient of R(X) proceeding from the left and call its corresponding power α.

Step 5. If $\alpha > S^1$ then a left slip of magnitude $\alpha - S^1$ has occured. If $\alpha < S^1$ then a right slip of $S^1 - \alpha$ places has occured. Finally if $\alpha = S^1$ no slip has occured.

Step 6. Multiply R(X) by $\bar{X}^{\alpha}$. Then $\bar{X}^{\alpha}R(X)$ is the transmitted n^1-tuple.

If we transmit words from V^1 then our expressions for slip are the same has 2.3.11. and 2.3.20. except that n becomes n^1. Rewriting these we have

$$X^L B(X) + \left[a_{n^1-L} + a_{n^1-1} X^{L-1}\right] + X^{n'} \left[b_{n^1-L} + \cdots b_{n^1-1} X^{L-1}\right] \qquad 4.2.9.$$

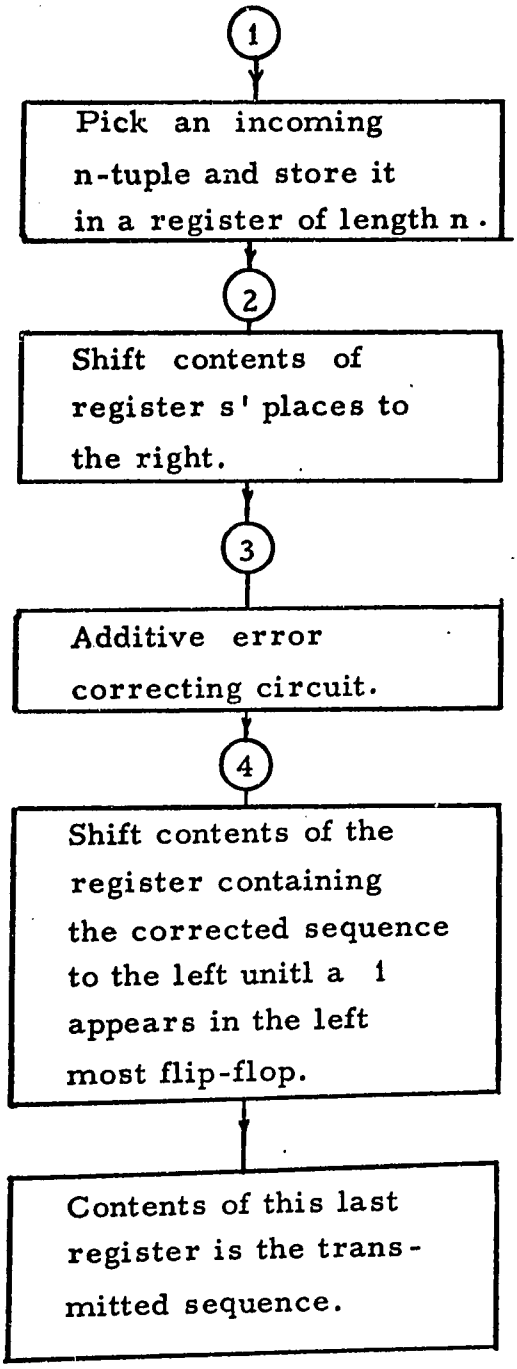
for a left slip and

$$\bar{X}^R B(X) + \bar{X}^R\left[a_0 + a_1 X + \ldots a_{R-1} X^{R-1}\right] + X^{n^1-R}\left[c_0 + c_1 X + \ldots c_{R-1} X^{R-1}\right] \qquad 4.2.10.$$

for a right slip.

Flow Chart For The Decoding Algorithm

Described on Page 59



In this case since we are decoding using V, $X^{n^1} \neq 1$ modulo X^n+1.

If we define

$$B_{L,1}(X)=a_{n^1-L}+\ldots a_{n^1-1}X^{L-1} \qquad 4.2.11.$$

$$B_{L,2}(X)=b_{n^1-L}+\ldots b_{n^1-1}X^{L-1} \qquad 4.2.12.$$

$$B_{R,1}(X)=a_0+a_1X+\ldots a_{R-1}X^{R-1} \qquad 4.2.13.$$

$$B_{R,2}(X)=c_0+c_1X+\ldots c_{R-1}X^{R-1} \qquad 4.2.14.$$

then 4.2.9. and 4.2.10. become

$$X^L B(X) + B_{L,1}(X) + X^{n^1} B_{L,2}(X) \qquad 4.2.15.$$

and $$X^{-R} B(X) + X^{-R} B_{R,1}(X) + X^{n^1-R} B_{R-2}(X) \qquad 4.2.16.$$

If we now follow our decoding algorithmn we have in Step 1

$$Y(X)=X^L B(X) + B_{L,1}(X) + X^{n^1} B_{L,2}(X)$$

or $$Y(X)=X^{-R} B(X) + X^{-R} B_{R,1}(X) + X^{n^1-R} B_{R,2}(X)$$ according as the slip is to the left or to the right respectively.

In Step 2 we will have

$$X^{S^1} Y(X) = X^{L+S^1} B(X) + X^{S^1} B_{L,1}(X) + X^{n^1+S^1} B_{L,2}(X) \qquad 4.2.17.$$

or $$X^{S^1} Y(X) = X^{S^1-R} B(X) + X^{S^1-R} B_{R,1}(X) + X^{n^1+S-R} B_{R,2}(X) \qquad 4.2.18.$$

In Step 3 we decode $X^{S^1} Y(X)$. In the case of a left slip we observe that:

1) $X^{L+S^1} B(X)$ is a word of V since V is cyclic.

2) $X^{S^1} B_{L,1}(X)$ is a burst of length S^1 or less and occuring

in the first $2S^1$ places.

3) $X^{n+S^1}B_{L,2}(X)$ is a burst of length S^1 or less occurring in the last $2S^1$ places. This last statement is true since the maximum power of $X^{n^1+S^1}B_{L,2}(X)$ is n^1+2S^1-1 which is, by our choice of n^1, less than n.

Therefore $X^{S^1}Y(X)$ is correctable by V.

In the case where a right slip has occured we observe that:

1) $X^{S^1-R}B(X)$ is a word of V.

2) $X^{S^1-R}B_{R,1}(X)$ is a burst of length S^1 or less occurring in the first $2S^1$ places. This is true since $S^1 \geq R$.

3) $X^{n^1+S^1-R}B_{R,2}(X)$ is a burst of length S^1 or less occurring in the last $2S^1$ places.

Hence once more $X^{S^1}Y(X)$ is correctable by V. Having established that $X^{S^1}Y(X)$ is correctable by V provided $0 \leq L,R \leq S^1$ we go to Step 4.

After correcting $X^{S^1}Y(X)$ we have

$$R(X) = X^{L+S^1}B(X)$$

or $R(X) = X^{S^1-R}B(X)$. Now since B(X) belongs to V^1 it starts with a 1. Secondly since $X^{L+S^1}B(X)$ and $X^{S^1-R}B(X)$ do not overflow modulo X^n+1 or their degree does not exceed n-1, it follows that the first non-zero coefficient of R(X) is $L+S^1$ in the case of a left slip and S^1-R in the case of a right slip. Calling this power α it follows that

$$L = \alpha - S^1$$

and $R = S^1 - \alpha$

In Step 5 we multiply $R(X)$ by $X^{-\alpha}$ to obtain

$B(X)$ in both cases.

Therefore we have shown that if V^1 is used for transmission and V for decoding the V^1 can correct all slips not exceeding S^1 where $S^1=(n-n^1)/2$.

QED

In Theorem 4.2.2. we have used a hypothetical code V which has the property that it can correct a burst of length S^1 or less always occurring in the first and last $2S^1$ places. If such a code can be constructed then the technique of Theorem 4.2.2. is efficient and easy to decode. In any case it is always possible to treat the burst errors as random errors and use for V a BCH code, instead of V.

COROLLARY 4.2.1.

If in Theorem 4.2.2. we use for V a t-random error correcting cyclic code, such as the BCH codes then V^1 can correct S^1 or less slip errors where

$$S^1 \leq (d-1)/4. \qquad 4.2.19.$$

Furthermore $d=2t+1$.

Proof: The two bursts in Theorem 4.2.2. are now treated as a random error of weight $2S^1$ or less. Therefore

$$2S^1 \leq \frac{d-1}{2}$$

or $S^1 \leq (d-1)/4$

QED

EXAMPLE 4.2.1.

In Corallory 4.2.1. let V be the (15,7,2) code generated by

$$G(X)=1+X^4+X^6+X^7+X^8.$$ (see Table 3.3.1.)

Since d=5, S^1 is bounded by 1 according to inequality 4.2.19. Therefore choose $S^1=1$ which implies that $n^1=15-2=13$. Therefore V^1 is a (13,4,0,1) code.

Let us choose from V^1 the words

$$G(X)(1+X+X^4) = 1+X+X^5+X^6+X^8+X^9+X^{10}+X^{11}+X^{12}$$

and $$G(X)(1+X^3) = 1+X^3+X^4+X^6+X^8+X^9+X^{10}+X^{11}.$$

Consider the incoming sequence

$$110001101111[1,100110101111]0.$$

Step 1. Pick from the sequence the term between square brackets, or

$$Y(X)=1+X+X^4+X^5+X^7+X^9+X^{10}+X^{11}+X^{12}$$

Step 2.

$$XY(X)=X+X^2+X^5+X^6+X^8+X^{10}+X^{11}+X^{12}+X^{13}$$

Step 3.

Decode XY(X) to find the error pattern

$$e(x)=X$$

$$R(X)=XY(X)+X=X^2+X^5+X^6+X^8+X^{10}+X^{11}+X^{12}+X^{13}.$$

Step 4.

The first non-zero coefficient of R(X) has power

Step 5.

Since $\alpha=2>1$ then a left shift of vdlue $\alpha-S^1=2-1=1$ has occurred.

Step 6.

Multiply $R(X)$ by $X^{-\alpha}=X^{-2}$ to obtain

$$X^{-2}R(X)= 1+X^3+X^4+X^6+X^8+X^9+X^{10}+X^{11}=G(X)(1+X^3)$$

or the desired code word.

END OF EXAMPLE

In the decoding algorithm given in Theorem 4.2.2. Step 3 invloves going through the existing decoding procedure of V for additive errors. What we are essentially doing is treating the bursts introduced by the slip error as additive errors. This step, of course, can be very involved as, for example, the decoding procedure for BCH codes has given by Peterson [13]. We can easily modify our decoding technique in such a way that we can use Fire codes instead of BCH codes. This will lead to a more efficient code since Fire codes are specifically designed to correct burst errors. Furthermore if we introduce on extra constraint an S^1 Step 3 can be done in one step.

Theorem 4.2.3.

Let V be an (n,k) cyclic code which can correct one burst of length $3S^1$ always occurring in the first $4S^1$ places. Let V^1 be the shortened (n^1,k^1) version of V and consisting

of only those words of V which start with 1. Then if V^1 is used for transmission and V for decoding and provided a certain decoding algorithm is used, V^1 can correct all slips not exceeding S^1. Furthermore $n^1=n-2S^1$ and $k^1=k-2S^1-1 \geq 0$.

Proof: As in the proof of Theorem 4.2.2. we start with the decoding algorithm.

DECODING PROCEDURE

Step 1. Pick from the incoming sequence an n^1-tuple $Y(X)$.

Step 2. Multiply $Y(X)$ by X^{3S^1}. This is equivalent to shifting to the right by $3S^1$ places in a shift register of length n.

Step 3. Decode $X^{3S^1}Y(X)$ using the existing decoding procedure of V. Call the corrected vector $R^1(X)$.

Step 4. Multiply $R^1(X)$ by X^{-2S^1} and call it $R(X)$.

Step 5. Locate the first non-zero coefficient in $R(X)$ and call its corresponding power α.

Step 6. If $\alpha>S^1$ then a left slip of $\alpha-S^1$ places as occurred. If $S^1>\alpha$ then a right slip of $S^1-\alpha$ places has occurred. Finally if $\alpha=S^1$ then no slip has taken place.

Step 7. Multiply $R(X)$ by $X^{-\alpha}$ to obtain the transmitted word.

As can be seen this decoding algorithm is identical with that of Theorem 4.2.2. except for steps 2,3, and 4.

After step 2 we will have, after a slight modification of

expressions, 3.2.17. and 3.2.18.

$$X^{3S^1}Y(X) = X^{3S^1+L}B(X) + X^{3S^1}B_{L,1}(X) + X^{n^1+2S^1+S^1}B_{L,2}(X) \qquad 4.2.20.$$

$$\text{or } X^{3S^1}Y(S) = X^{3S^1-R}B(X) + X^{3S^1-R}B_{R,1}(X) + X^{n^1+2S^1+S^1-R}B_{R,2}(X) \qquad 4.2.21.$$

according as the slip is to the left or to the right respectively.

Now replacing n^1+2S^1 by n and realizing that X^n modulo X^n+1 is one, expressions 4.2.20. and 4.2.21. become

$$X^{3S^1}Y(X) = X^{3S^1+L}B(X) + X^{S^1}B_{L,2}(X) + X^{2S^1}B_{L,1}(X) \qquad 4.2.22.$$

and

$$X^{3S^1}Y(X) = X^{3S^1-R}B(X) + X^{S^1-R}B_{R,2}(X) + X^{3S^1-R}B_{R,1}(X) \qquad 4.2.23.$$

respectively.

In 4.2.22. we note that:

1) $X^{3S^1+L}B(X)$ is a word of V,
2) $X^{S^1}B_{L,2}(X) + X^{2S^1}B_{L,1}(X)$ is a burst of length $3S^1$ or less always occurring in the first $4S^1$ places.

In 4.2.23. we also note that:

1) X^{3S^1-R} is a code word of V,
2) $X^{S^1-R}B_{R,2}(X) + X^{3S^1-R}B_{R,1}(X)$ is a burst of length $3S^1$ always occurring in the first $4S^1$ places.

Therefore $X^{3S^1}Y(X)$ is decodable by V.

In Step 3 after correction we will have

$$R^1(X) = X^{3S^1+L}B(X)$$

or

$$R^1(X) = X^{3S^1-R}B(X).$$

In Step 4 after multiplication by X^{-2S^1} we have

$$R(X) = X^{S^1+L} B(X)$$

or $$R(X) = X^{S^1-R} B(X).$$

From this point on the situation is identical to Theorem 4.2.2. Hence V^1 does indeed correct all slips not exceeding S^1.

QED

This trivial modification of Theorem 4.2.2. therefore allows us to use Fire codes in a very elegant way. For an exposé of Fire codes see Peterson [1] Chapter 10.

COROLLARY 4.2.2.

In Theorem 4.2.3. let V be an (n,k) Fire code generated by G(X), a polynomial of degree n-k, which can correct any burst of length b or less. Then V^1 can correct all slips not exceeding S^1 where

$$S^1 = [b/3]^* \qquad 4.2.24.$$

Furthermore if $4S^1 \leq n-k$ then Step 3 can be done in one step.

Proof: Since in Theorem 4.2.3., V has to correct a burst of length $3S^1$ then it is immediate that $3S^1 \leq b$ or

$$S^1 \leq b/3.$$

Secondly if $4S^1 \leq n-k$ and since the burst always occurs in the first $4S^1$ places then the decoding procedure will take only

* $[b/3] = q$ where $b = 3q + r$ and $0 \leq r < 3$.

one step. Actually the error pattern of Step 3 in the decoding procedure of Theorem 4.2.3. will simply be the remainder of $X^{3S^1}Y(X)$ modulo $G(X)$. This is a property of Fire codes. For details see Peterson [1] Chapter 2 and reference [32].

QED

EXAMPLE 4.2.2.

Let V be the Fire code generated by $G(X)=(1+X+X^2+X^3+X^4)$ $(1+X^3) = 1+X+X^2+X^5+X^6+X^7$.

This is a (15,8) binary cyclic code which can correct any burst of length 4 or less.

By 4.2.24. S^1 is bounded by $b/3=4/3=11/4$ and $k^1=8-2-1=5$. Hence $n^1=n-2S^1=15-2=13$ and $k^1=9-2-1=5$. Let us choose from V^1 two words

$$G(X)(1+X^2+X^5) = 1+X+X^3+X^4+X^7+X^8+X^9+X^{10}+X^{11}+X^{12}$$

and

$$G(X)(1+X^4) = 1+X+X^2+X^4+X^6+X^{10}+X^{11}+X^{12}.$$

Consider the incoming sequence

$$1\ [101100111111,1]\ 1101010001110.$$

Step 1. Pick from the sequence the polynomial between the square brackets. Therefore

$$Y(X)= 1+X^2+X^3+X^6+X^7+X^8+X^9+X^{10}+X^{11}+X^{12}$$

Step 2. Multiplying by $X^{3S^1}= X^3$ gives,

$$X^3Y(X)= X^3+X^5+X^6+X^9+X^{10}+X^{11}+X^{12}+X^{13}+X^{14}+X^{15}$$

and since $n=15$, $+X^{15}=1$ or

$$X^3Y(X)= 1+X^3+X^5+X^6+X^9+X+X^{10}+X^{11}+X^{12}+X^{13}+X^{14}.$$

Step 3. Decode $X^3Y(X)$. Since $4S^1=4$ n-k= 7 we can do this in one step. Dividing $X^3Y(X)$ by G(X) we obtain as remainder $1+X^2$, which is the error pattern. Adding $1+X^2$ to $X^3Y(X)$ we obtain

$$R^1(X)= X^2+X^3+X^5+X^6+X^9+X^{10}+X^{11}+X^{12}+X^{13}+X^{14}.$$

Step 4. Multiplying $R^1(X)$ by $X^{-2S^1}=X^{-2}$ we have

$$X^{-2}R^1(X)=R(X)=1+X+X^3+X^4+X^7+X^8+X^9+X^{10}+X^{11}+X^{12}$$

Step 5. The first non-zero coefficient in R(X) has power $\alpha=0$.

Step 6. Since $\alpha<S^1=1$, we conclude that a right slip of $S^1-\alpha=1-0=1$ place has occurred.

Step 7. Multiplying R(X) by $X^{-\alpha}=1$ we have the transmitted code word

$$G(X)\ 1+X^2+X^5 =1+X+X^3+X^4+X^7+X^8+X^9+X^{10}+X^{11}+X^{12}$$

This completes the Example.

In Example 4.2.1. we started with a (15,7,2) BCH code and we constructed a (13,4,0,1) code which could correct 1 slip error. The transmission rate of V^1 in this case is 4/13=.308. In Example 4.2.2. we started with a (15,8) Fire code which can correct a single burst of length 3. From this Fire code we constructed a (13,5) code which could correct 1 slip error. The transmission rate of V^1 in this case is 5/13=.385. What is even more important than the transmission rate is the fact that in Example 4.2.2. we could correct the

additive error in one step. On the other hand in Example 4.2.1. we would have to go through the regular decoding procedure for BCH codes. This striking feature is of course expected since Fire codes are burst error correcting codes.

We have started with a very naive idea in Theorem 4.2.1. and by slight modifications we have been able to arrive at some important results. We are able at this point to modify our idea once more. In Theorem 4.2.2. we used a shortened version of the parent code V for transmission. This shortening process is identical with shortening the information k-tuple to a k^1-tuple where $k^1=k-2S^1-1$. This latter process is equivalent to making the last $2S^1$ information symbols identically zero. The 1 comes in because the words must start with a 1. If the information k-tuple has its last $2S^1$ symbols identically zero then the corresponding n-tuple will also have its last $2S^1$ symbols equal to zero. So what we do in Theorem 4.2.2. is to drop the last $2S^1$ zeros in the code words before transmission. However if we transmit the zeros we obtain a slightly different result.

Theorem 4.2.4.

Let V be an (n,k) binary cyclic code which can correct a single burst of length S^1 or less always occurring in the first $2S^1$ places. Then there exists a subset of V, say V^1, which can correct all slip errors not exceeding S^1 provided a certain decoding procedure is followed. Again S^1 is

bounded by $(k-1)/2$.

Proof: The decoding algorithm is the same as given in Theorem 4.2.2. remembering that X^n modulo X^n+1 is 1.

The subset V^1 is made up of all code words of V which start with a 1 and terminate with $2S^1$ or more zeros.

The expressions for slip are the same as in 4.2.18. except that n^1 is replaced by n. Furthermore X^n is replaced by 1. Therefore we have

$$X^{S^1}Y(X) = X^{S^1+L}B(X) + X^{S^1}B_{L,1}(X) + X^{S^1}B_{L,2}(X) \qquad 4.2.25.$$

and $$X^{S^1}Y(X) = X^{S^1-R}B(X) + X^{S^1-R}B_{R,1}(X) + X^{S^1-R}B_{R,2}(X).$$

These can be rewritten as

$$X^{S^1}Y(X) = X^{S^1+L}B(X) + X^{S^1}U_L(X)$$

and $$X^{S^1}Y(X) = X^{S^1-R}B(X) + X^{S^1-R}U_R(X)$$

where $U_L(X) = B_{L,1}(X) + B_{L,2}(X)$

and $U_R(X) = B_{R,1}(X) + B_{R,2}(X)$

It is immediate that $X^{S^1}Y(X)$ is correctable by V.

Therefore in Step 3 the corrected n-tuple is

$X^{S^1+L}B(X)$

or $X^{S^1-R}{}_{3(X)}$ according as the slip is to the left or to the right respectively.

Now since B(X) belongs to V^1, its maximum degree is $n-2S^1-1$ or as in Theorem 4.2.2. the first non-zero coefficient term

in Step 4 has power S^1+L or S^1-R. As in Theorem 4.2.2., the technique always works. The only difference in this case is that the parent code has only to correct one burst instead of 2. On the other hand the transmission rate in this case is lower.

QED

There are as before two immediate Corallories to Theorem 4.2.9.

COROLLARY 4.2.3.

If in Theorem 4.2.9. V is an (n,k,t) binary cyclic code then V^1 can correct all slip errors not exceeding S^1 where

$$S^1 \leq (d-1)/2 \qquad 4.2.26.$$

and $d=2t+1$.

Proof: A burst of length S^1 can be considered as a random error of weight S^1 or less.

QED

COROLLARY 4.2.4.

If in Theorem 4.2.5. V is an (n,k) cyclic burst error correcting code capable of correcting a single burst of length b, then V^1 can correct all slips not exceeding S^1 where

$$S^1 < b. \qquad 4.2.27.$$

Furthermore if $2S^1 \leq n-k$ the degree of the generator polynomial of V, then Step 3 in the algorithm given in Theorem 4.2.2.

takes only one step.

Proof: Same as for Corollary 4.2.2.

QED

EXAMPLE 4.2.3.

In Corollary 4.2.4. let V be the Fire code generated by $G(X)= 1+X+X^2+X^5+X^6+X^7$. V can correct any burst of length 3 or less or b=3. Furthermore n=15 and k=8.

In this case since $S^1 \leq b=3$, we can correct three slip errors. Since $\frac{k-1}{2} = 7/2$ we are all right. Hence $k^1=k-2S^1-1=8-7=1$ or there are only two words in the code. Since this would be a trivial example let us make $S^1=2$. In this case $k^1=8-5=3$.

Choose from V^1,

$$G(X)\ (1+X+X^3)= 1+X^4+X^9+X^{10}$$

and $$G(X)\ (1+X^2) = 1+X+X^3+X^4+X^5+X^6+X^8+X^9$$

and consider the sequence

$$10[0010000110000,11]0111101100000.$$

Step 1. Pick from the sequence the term between the square brackets or

$$Y(X)= X^2+X^7+X^8+X^{13}+X^{14}$$

Step 2. Multiply Y(X) by $X^{S^1}=X^2$ to obtain

$$X^2Y(X)= X^4+X^9+X^{10}+X^{15}+X^{16}$$

or $$X^2Y(X)= 1+X+X^4+X^9+X^{10}$$

since $X^{15+8}=X^8$.

Step 3. Decode $X^2Y(X)$! Since $S^1=2<n-k=7$ we can do this in one step. Dividing $X^2Y(X)$ by G(X) we obtain X as the remainder. After correction,

$$R(X)=X^2Y(X) + X^2 = 1+X^4+X^9+X^{10}$$

Step 4. The first non-zero term in R(X) has power $a=0$.

Step 5. Since $a=0$ $S^1=2$ we conclude that a right slip of magnitude $S^1 - a = 2-0=2$.

Step 6. Multiplying R(X) by $X^{-a}=1$ we have the transmitted word $1+X^4+X^9+X^{10}$.

The example is completed.

We now give two Tables of results obtained using the theory developed in this section.

1	2	3	4	5	6	7	8
				COR. 4.2.1.		COR. 4.2.3.	
(n,k,t)	G(X)	k/n	S^1	(n^1,k^1)	k^1/n^1	(n,k)	k^1/n^1
(15,11,1)	23	.734	1			(15,8)	.533
(15,7,2)	721	.467	1	(13,4)	.308	(15,4)	.267
			2			(15,2)	.133
(31,21,2)	3551	.627	1	(29,18)	.620	(31,18)	.580
			2			(31,16)	.516
(31,16,3)	107,657	.517	1	(29,13)	.448	(31,13)	.420
			2			(31,11)	.355
			3			(31,9)	.290
(127,92,5)	6247300 31467	.725	1	(125,89)	.712	(127,89)	.700
			2	(123,87)	.706	(127,87)	.685
			3			(127,85)	.670
			4			(127,83)	.653
			5			(127,81)	.636
(127,64,10)		.505	1	(125,61)	.488	(127,61)	.480
			2	(123,59)	.480		
			3	(121,57)	.470		
			4	(119,55)	.462		
			5	(117,53)	.453		
			⋮	⋮	⋮	⋮	⋮
			10			(127,43)	.339
(255,163,12)		.640	1	(253,160)	.632	(255,160)	.627
			6	(243,150)	.617	(255,150)	.589
			⋮	⋮	⋮	⋮	⋮
			12	(		(255,138)	.541

TABLE 4.2.1.

LEGEND TO TABLE 4.2.1.

Column 1. The parent code V. t is the additive error correcting capability of V.

Column 2. G(X) is the generator polynomial of the BCH code. G(X) tabulated as explained in Table 3.3.1.

Column 3. k/n is the transmission rate of V.

Column 4. S^1 is the slip error correcting capability of V.

Column 5. (n^1,k^1) of V^1 has given in Corollary 4.2.1. That is to say $n^1=n-2S^1$ and $k^1=n-2S^1-1$. The slash indicates that V^1 does not exist or S^1 $(d-1)/4$ as given by 4.2.19.

Column 6. k^1/n^1 is the transmission rate of V^1 as given in Column 5.

Column 7. (n^1,k^1) of V^1 has given by Corollary 4.2.3. That is to say $n^1=n$, $k^1=k-2S^1-1$. The entries stop when $S^1=(d-1)/2$ as given by 4.2.26.

Column 8. k^1/n^1 is the transmission rate of V^1 as given in Column 7.

1	2	3	4	5	6	7	8	9
				COR', 4.2.2.			COR. 4.2.4.	
(n, k)	b	$G(X)=p(x)(1+x^c)$	k/n	S^1	(n^1,k^1)	k^1/n^1	(n^1,k^1)	k^1/n^1
(15, 8)	4	$(1+X+X^2+X^3+X^4)$ $(1+X^3)$	.535	1	(13, 5)		(15, 5)	.333
				2			(15, 3)	.200
(21, 8)	6	$(1+X+X^2+X^4+X^6)$ $(1+X^7)$	.380	1	(19, 5)	.263	(21, 5)	.238
				2	(17, 3)	.176	(21, 3)	.143
(56, 39)	9	$(1+X^3+X^4+X^7$ $+X^9)(1+X^8)$	.697	1	(54, 36)	.667	(56, 36)	.643
				2	(52, 34)	.655	(56, 34)	.608
				3	(50, 32)	.640	(56, 32)	.572
				4			(56, 30)	.536
				:	:::	:	:::	:
				9			(56, 20)	.358
(55, 22)	12	$(1+X^2+X^3+X^6$ $+X^{12})(1+X^{11})$	.400	1	(53, 19)	.359	55, 19)	.346
			.	.	...	.	...	.
				4	(47, 13)	.277	(55, 13)	.237
			.	.	...	.	...	.

TABLE 4.2.2.

LEGEND TO TABLE 4.2.2.

Column 1. (n,k) of the parent Fire Code V.

Column 2. b the burst error correcting capability of V.

Column 3. G(X) is the generator polynomial of V.

Column 4. k/n is the transmission rate of V.

Column 5. S^1 is the slip error.

Column 6. (n^1,k^1) of the code V^1 has given in Corollary 4.2.2. That is to say $n^1=n-2S^1$ and $k^1=k-2S^1-1$.

Column 7. k^1/n^1 is the transmission rate of V^1 given in Column 6.

Column 8. (n^1,k^1) of V^1 as given by Corollary 4.2.4. That is to say $n^1=n$ and $k^1=k-2S^1-1$.

Column 9. k^1/n^1 is the transmission rate of V^1 as given in Column 8.

4.3. NOISY CASE

The noiseless case discussed earlier can be easily extended to the noisy case. The expressions for slip are not much different, as was seen in Chapter 3 section 2. We start by giving the central theorem of this chapter.

THEOREM 4.2.5.

Let V be an (n,k) binary cyclic code which can correct a burst of length S^1 or less always occurring in the first $2S^1$ places, a second burst of length S^1 or less always occurring in the last $2S^1$ places, and a third error E(X) occurring in between. The error E(X) belongs to a class of errors E which need not be defined for the moment. Let V^1 be the shortened (n^1,k^1) version of V, consisting of only those words which start with a 1. Then if V^1 is used for transmission and V for decoding, and provided a certain decoding algorithm is followed, V^1 is an E(X) synchronizable error correcting code provided the slip does not exceed S^1 and where the additive error per word is E(X). Furthermore S^1 is bounded by (k-1)/2 and $n^1=n-2S^1$, $k^1=k-2S^1-1$.

Proof: The decoding algorithm is identical with the one given in Theorem 4.2.2.

If an additive error E(X) is allowed to corrupt each code word transmitted, then the expressions for left and right slip become

$$X^L B(X) + B_{L,1}(X) + X^{n^1} B_{L,2}(X) + E_1(X) \qquad 4.2.28$$

and

$$X^{-R} B(X) + X^{-R} B_{R,1}(X) + X^{n^1-R} B_{R,2}(X) + E_2(X) \qquad 4.2.29$$

respectively and where $E_i(X)$ belongs to E.

Let us now follow through the decoding algorithm given in Theorem 4.2.2. but using our new set of equations.

In Step 2 after multiplication by X^{S^1} we will have;

$$X^{S^1}Y(X) = X^{S^1+L}B(X) + X^{S^1}B_{L,1}(X) + X^{n^1+S^1}B_{L,2}X + X^{S^1}E_1(X) \qquad 4.2.30.$$

or

$$X^{S^1}Y(X) = X^{S^1-R}B(X) + X^{S^1-R}B_{R,1}(X) + X^{n^1+S^1-R}B_{R,2}(X) + X^{S^1}E_2(X). \qquad 4.2.31.$$

Assuming that $X^{S^1}E_1(X)$ and $X^{S^1}E_2(X)$ belong to E, we note that 4.2.30. and 4.2.31. are both correctable by V provided $0 \leq L, R \leq S^1$. After correction we will have

$$R(X) = X^{S^1+L}B(X) \qquad 4.2.32.$$

or

$$R(X) = X^{S^1-R}B(X) \qquad 4.2.33.$$

Since 4.2.32 and 4.2.33 are identical with the expressions obtained in Step 3 of Theorem 4.2.2. all the arguments used there apply equally well here. Hence we have completed the proof of our Theorem. QED

We should perhaps mention that in Theorem 4.2.5. V need not be a random error correcting code, It is in fact a lot more general than this. It is felt that if a random error correcting code is used then the technique will be less efficient.

However if E(X) is a random error and if the slip error is small compared to the additive error then a purely random error correcting code becomes efficient. In the limit when the slip error is only 1, the bursts introduced by the slip error each become of length 1 or less. In this case the two brusts can be considered as a random error of weight 2 or less. Some work has been done on multiple burst error correcting codes. The work done so far does not give a class of codes for which there exists a decoding procedure. For example see the two papers by Stone [33,34] .

As applied to random error correcting codes we have,

COROLLARY 4.2.4.

In Theorem 4.2.5. if V is a t random error correcting cyclic code, and if E is the class of random errors of weight e or less, where $e<t$ then V^1 is a synchronizable e single digit error correcting code provided the slip does not exceed S^1. Furthermore,

$$S^1 \leq (d-2e-1)/4 \qquad 4.2.34.$$

Proof: In Theorem 4.2.5., we now consider the two bursts of length S^1 or less as a random error. Hence this will be a random error of weight $2S^1$ or less. Furthermore E(X) is now an additive error of weight e or less. Hence altogether the error

has weight $2S^1+e$ or less. Hence since V is now a t random error correcting code, we must have $2S^1+e \leq (d-1)/2$ where $d=2t+1$.

Equivalently, $S^1 \leq (d-2e-1)/4$.

QED

EXAMPLE 4.2.4.

Let V in Corollary 4.2.4. be the (31,16,3) BCH code generated by

$$G(X)=1+X+X^2+X^3+X^5+X^7+X^8+X^9+X^{10}+X^{11}+X^{15}.$$

According to 4.2.24.

$S^1 \leq (d-2e-1)/4$, hence if we choose e=1, $S^1 \leq 7-3/4=1$,

Therefore V^1 will be a (29,13,1,1) code.

Choose from V^1 the words

$$W_1(X)=G(X)(1+X^2+X^{12}) = 1+X+X^9+X^{14}+X^{19}+X^{20}+X^{21}+X^{22}+X^{23}+X^{27},$$

and $W_2(X)=G(X)(1+X^{13}) = 1+X+X^2+X^3+X^5+X^7+X^8+X^9+X^{10}+X^{11}+X^{13}+X^{14}$
$+X^{16}+X^{18}+X^{20}+X^{21}+X^{22}+X^{23}+X^{24}+X^{28}.$

Let the additive error in $W_1(X)$ be X^{28} and in $W_2(X)$ be 1 and consider the sequence $W_1(X)+X^{28}$, $W_2(X)+1$, or

$$1[1001000000000100001111100011,0]11101011111011010101111100001.$$

In this example we will work out the case of a left and a right slip.

A. Right Slip.

Pick the sequence contained in the square brackets, that is to say:

$$Y(X)=1+X^3+X^{13}+X^{18}+X^{19}+X^{20}+X^{21}+X^{22}+X^{26}+X^{27}.$$

Step 2. We multiply Y(X) by $X^{S^1}=X$ to obtain

$$Y(X)=X+X^4+X^{14}+X^{19}+X^{20}+X^{21}+X^{22}+X^{23}+X^{27}+X^{28}.$$

Step 3. We decode XY(X) using the procedure for decoding BCH code for V. Doing this we obtain the error pattern

$E(X) = 1$. Correcting XY(X) we have

$$R(X)=XY(X)+1=1+X+X^4+X^{14}+X^{19}+X^{20}+X^{21}+X^{22}+X^{23}+X^{27}+X^{28}.$$

Step 4. The first non-zero coefficient of R(X) has power $\alpha=0$.

Step 5. Since $\alpha=0<S^1=1$ we conclude that a right slip of magnitude $S^1-\alpha=1-0=1$ has occurred.

Step 6. Multiplying R(X) by $X^{-\alpha}=1$ we have the transmitted word:

$$W(X)= 1+X+X^4+X^{14}+X^{19}+X^{20}+X^{21}+X^{22}+X^{23}+X^{27}+X^{28}.$$

B. Left Slip

Step 1. Pick from the sequence the expression contained within the round brackets. Therefore we have

$$Y(X)= 1+X^2+X^3+X^4+X^6+X^8+X^9+X^{10}+X^{11}+X^{12}+X^{14}+X^{15}+X^{19}+X^{21}$$
$$+X^{22}+X^{23}+X^{24}+X^{25}$$

Step 2. Multiplying Y(X) by $X^{S^1}=X$ we obtain

$$XY(X)= X+X^3+X^4+X^5+X^7+X^9+X^{10}+X^{11}+X^{12}+X^{13}+X^{15}+X^{16}+X^{18}+X^{20}$$
$$+X^{22}+X^{23}+X^{24}+X^{25}+X^{26}$$

Step 3. Decoding XY(X) we obtain

$$E(X)= X+X^2+X^{30} \text{ from which we obtain}$$

$$R(X)=XY(X)+E(X)=X^2+X^3+X^4+X^5+X^7+X^9+X^{10}+X^{11}+X^{12}+X^{13}+X^{15}+X^{16}$$
$$+X^{18}+X^{20}+X^{22}+X^{23}+X^{24}+X^{25}+X^{26}+X^{30}.$$

Step 4. The first non-zero coefficient in R(X) has power $\alpha=2$.

Step 5. Since $\alpha=2 > S^1=1$, we conclude that a left slip of magnitude $\alpha-S^1= 2-1=1$ places has occurred.

Step 6. Multiplying R(X) by $X^{-\alpha}=X^{-2}$ we have the transmitted code word

$$W_2(X)=X^{-2}R(X)=1+X+X^2+X^3+X^5+X^7+X^8+X^9+X^{10}+X^{11}+X^{13}+X^{14}+X^{16}$$
$$+X^{18}+X^{20}+X^{21}+X^{22}+X^{23}+X^{24}+X^{28}.$$

This completes the example.

Since it is probably easier to construct a code which can correct 1 burst and an error E(X) than it is to construct a code capable of correcting 2 bursts and a third error E(X) we state a modification of Theorem 4.2.5. This next Theorem is analogous to Theorem 4.2.3. in the noiseless case.

Theorem 4.2.6.

Let V be an (n,k) binary cyclic code which can correct a burst of length $3S^1$ always occurring in the first $4S^1$ places

and an error E(X) occurring somewhere else. E(X) belongs to a class of errors E which need not be defined for now. Let V^1 be the shortened (n^1,k^1) version of V and consisting of only those words starting with a 1. Then if V^1 is used for transmission and V for decoding V^1 is an E(X) synchronizable error correcting code provided the slip is no larger than S^1 and provided a certain decoding algorithm is followed. Furthermore $n^1=n-2S^1$ and $k^1=k-2S^1-1$ where $S^1 \leq (k-1)/2$.

Proof: The decoding algorithm is the same as the one given in Theorem 4.2.3.

Step 1. In this case the equations will be

$$Y(X)= X^L B(X) + X^{n^1} B_{L,2}(X) + B_{L,1}(X) + E_1(X) \qquad 4.2.34.$$

or $$Y(X)= X^{-R} B(X) + X^{-R} B_{R,1}(X) + X^{n^1-R} B_{R,2}(X) + E_2(X) \qquad 4.2.35.$$

according as the slip is to the left or to the right respectively and $E_i(X)$ belongs to E i=1,2.

Step 2. After multiplication by X^{3S^1} we will have

$$X^{3S^1} Y(X)= X^{L+3S^1} B(X)+ X^{3S^1+n^1} B_{L,2}(X)+ X^{3S^1} B_{L,1}(X)+ X^{3S^1} E_1(X)$$

and $$X^{3S^1} Y(X)= X^{3S^1-R} B(X)+ X^{3S^1-R} B_{R,1}(X)+ X^{n^1+2S^1+S^1-R} B_{R,2}(X)$$
$$+ X^{3S^1} E_2(X).$$

Replacing n^1+2S^1 by n and X^n by 1 we have

$$X^{3S^1} Y(X)= X^{L+3S^1} B(X)+ X^{S^1} B_{L,2}(X)+ X^{2S^1} B_{L,1}(X) + X^{3S^1} E(X) \qquad 4.2.36.$$

and

$$X^{3S^1} Y(X)= X^{3S^1-R} B(X)+ X^{3S^1-R} B_{R,1}(X)+ X^{S^1-R} B_{R,2}(X)+ X^{3S^1} E_2(X). \qquad 4.2.37.$$

Assuming that $X^{3S^1}E_i(X)$ i=1,2 belongs to E, we realize that 4.2.36. and 4.2.37. are both correctable by V. After correction we have

$$R^1(X)= X^{L+3S^1}B(X) \qquad 4.2.38.$$

and

$$R^1(X)= X^{3S^1-R}B(X) \qquad 4.2.39.$$

Since these last two equations are indentical with those in Step 3 of Theorem 4.2.3. the proof follows.

QED

If E is a class of burst errors and if we make the restriction that an additive and a slip error do not occur simultaneously in a code word then a Fire code can be used to realize Theorem 4.2.6. Since a Fire code can be made to correct any burst of length b and simultaneously detect any burst of length d, then if a slip and an additive error occur simultaneously we could ask for retransmission. If we wish to use a random error correcting code then Theorem 4.2.5. is more efficiently implemented.

Our last Theorem in this section is one analogous to Theorem 4.2.9. in the noiseless case.

Theorem 4.2.7.

Let V be a binary cyclic code which can correct a single burst

of length S^1 or less always occurring in the first $2S^1$ places and an error $E(X)$ occurring elsewhere. $E(X)$ belongs to a class E of additive errors which need not be defined for now. Let V^1 be the subset of V consisting on only those words which start with a 1 and terminate with $2S^1$ or more zeros. Then if V^1 is used for transmission and V for decoding, V^1 can correct all slip errors not exceeding S^1 and the simultaneous occurence of an additive error $E(X)$ per transmitted word, provided a certain decoding algorithm is followed.

Finally, V^1 is an (n,k^1) code with $k^1=k-2S^1-1$.

Proof: The decoding procedure is the same one as used in Theorem 4.2.2.

Step 1. In this situation since $n^1=n$ and $X^n=1$,

$$Y(X)=X^L B(X) + B_{L,1}(X)+ B_{L,2}(X) + E_1(X) \qquad 4.2.40.$$

or

$$Y(X)=X^{-R}B(X) + X^{-R}B_{R,1}(X) + X^{-R}B_{R,2}(X) + E_2(X) \qquad 4.2.41.$$

according as the slip is to the left or to the right respectively.

Step 2. After multiplication by X^{S^1} and labelling $U_L(X)=B_{L,1}(X) + B_{L,2}(X)$ and $U_R(X)= U_{R,1}(X) + U_{R,2}(X)$ we have

$$X^{S^1}Y(X)= X^{L+S^1}B(X) + X^{S^1}U_L(X) + X^{S^1}E_1(X) \qquad 4.2.42.$$

or

$$X^{S^1}Y(X)= X^{S^1-R}B(X) + X^{S^1-R}U_R(X) + X^{S^1}E_2(X) \qquad 4.2.43.$$

Step 3. Observing that 4.2.42. and 4.2.43. are both correctable by V, provided $0 \leq L, R \leq S^1$ and $X^{S^1} E_i(X)$ belongs to E for i=1,2, we have after correction

$$R(X) = X^{L+S^1} B(X) \qquad 4.2.44.$$

or $$R(X) = X^{S^1-R} B(X) \qquad 4.2.45.$$

Since these last two expressions are identical with those obtained in Step 3 of theorem 4.2.4., the proof follows.

QED

COROLLORY 4.2.5.

In Theorem 4.2.7. if V is a t random error correcting binary cyclic code and if E is a class of random errors of weight e or less, then V^1 is capable of correcting all slip errors not exceeding S^1 and the simultaneous occurrence of e random errors per code word, where

$$e + S^1 \leq t \qquad 4.2.46.$$

and $$S^1 \leq (k-1)/2.$$

Proof: Since the burst introduced by the slip error is now considered as a random error of weight S^1 or less and E(X) is a random error of weight e or less, the total error has weight $e+S^1$ or less. Since V is a t random error correcting code it follows that we must have

$$e+S^1 \leq (d-1)/2 = t$$

or $s^1 \leq (d-2e-1)/2$.

QED

EXAMPLE 4.2.5.

In Corollary 4.2.5. let V be the (31,21,2) BCH code generated by $G(X)= 1+X^3+X^5+X^6+X^8+X^9+X^{10}$.

If we let e=1 then

$$s^1 \leq (d-2e-1)/2 = (5-2-1)/2 = 1.$$

Therefore V^1 is an (31,18,1,1) code.

Choose from V^1

$W(X)=G(X)(1+X^{16}+X^{17}) = 1+X^3+X^5+X^6+X^8+X^9+X^{10}+X^{16}+X^{17}+X^{19}$
$+X^{20}+X^{21}+X^{23}+X^{24}+X^{27}$

$W_2(X)=G(X)(1+X^{11}) = 1+X^3+X^5+X^6+X^8+X^9+X^{10}+X^{11}+X^{14}+X^{16}+X^{17}$
$+X^{19}+X^{20}+X^{21}$.

Let the error in $W_1(X)$ be X^{30} and in $W_2(X)$ be X^{15}. Consider the sequence $W_1(X) + X^{30}$, $W_2(X) + X^{15}$, which is

1 [00101101110000011011101100100(1,1] 00101101111001111011100000000)0,

A. Left Slip

Step 1. Pick the sequence within the round brackets or

$Y(X)=1+X+X^4+X^6+X^7+X^9+X^{10}+X^{11}+X^{12}+X^{15}+X^{16}+X^{17}+X^{18}+X^{20}+X^{21}+X^{22}$

Step 2. Multiplying Y(X) by $X^{s^1}=X$ we have

$XY(X)=X+X^2+X^5+X^7+X^8+X^{10}+X^{11}+X^{12}+X^{13}+X^{16}+X^{17}+X^{18}+X^{19}+X^{21}+X^{22}$
$+X^{23}$

Step 3. Decoding XY(X) we obtain $X+X^{17}$ as our error pattern. Correcting XY(X) we obtain

$$R(X)=X+X^{17}+XY(X)=X^2+X^5+X^7+X^8+X^{10}+X^{11}+X^{12}+X^{13}+X^{16}+X^{18}+X^{19}+X^{21}+X^{22}+X^{23}.$$

Step 4. The first non-zero coefficient in R(X) has power $\alpha=2$.

Step 5. Since $\alpha=2>S^1=1$ we conclude that a left slip of magnitude $\alpha-S^1=2-1$ has occurred.

Step 6. Multiplying R(X) by $X^{-\alpha}$ we obtain the transmitted word

$$X^{-2}R(X)=W_2(X)=1+X^3+X^5+X^6+X^8+X^9+X^{10}+X^{11}+X^{14}+X^{16}+X^{17}+X^{19}+X^{20}+X^{21}.$$

B. Right Slip

Step 1. Pick the sequence within the square brackets, or

$$Y(X)=X^2+X^4+X^5+X^7+X^8+X^9+X^{15}+X^{16}+X^{18}+X^{19}+X^{20}+X^{22}+X^{23}+X^{26}+X^{29}+X^{30}$$

Step 2. Multiplying Y(X) by $X^{S^1}=X$ we have

$$Y(X)=X^3+X^5+X^6+X^8+X^9+X^{10}+X^{16}+X^{17}+X^{19}+X^{20}+X^{21}+X^{23}+X^{24}+X^{27}+X^{30}.$$

Step 3. Decoding XY(X) we obtain X^{30} has the errors hence the corrected sequence is

$$XY(X)+X^{30}=R(X)=1+X^3+X^5+X^6+X^8+X^9+X^{10}+X^{16}+X^{17}+X^{19}+X^{20}+X^{21}+X^{23}+X^{24}+X^{27}.$$

Step 4. The first non-zero coefficient in R(X) has power $\alpha=0$.

Step 5. Since $\alpha=0<S^1=1$ we conclude that a right slip of

magnitude $S^{1-\alpha}=1-0=1$ has occurred.

Step 6. Multiplying R(X) by $X^{-\alpha}=1$ we obtained the transmitted code word

$$R(X)=W_1(X)=1+X^3+X^5+X^6+X^8+X^9+X^{10}+X^{16}+X^{17}+X^{19}+X^{20}+X^{21}+X^{23}+X^{24}+X^{27}.$$

QED

Comparing examples 4.2.4. and 4.2.5. we notice that for this particular situation anyway, Corollary 4.2.5. leads to a more efficient code than Corollary 4.2.4. In example 4.2.5. we have a (31,18,1,1) code with a transmission rate of .580 while in example 4.2.4. we arrived at a (29,13,1,1) code with transmission rate .495.

CHAPTER 5

CONCLUSIONS

In Chapter 3 we have described two major techniques to synchronize binary cylic codes. The main reason why we limit our attention to binary cyclic codes is because there exists a class of t-random error correcting binary cyclic codes for which there exists a decoding procedure. These are of course the BCH codes. In section 3.2. we have presented some theorems on the coset technique. The main theorem in this section being Theorem 3.2.3. in which the code developed can handle both additive and slip errors. Unfortunately there does not exist at the present time a decoding procedure for the coset codes. By a decoding procedure we exclude the possibility of storing all the possible residues and then using a search procedure at the decoder. Nevertheless the results on coset codes provide us with a good yardstick to measure the performance of other synchronizable-error-correcting codes. It is hoped that a decoding procedure will be developed for these codes. This in itself might be a good subject for a future Master's Thesis. The importance of coset codes is not to be overlooked because we do not have a decoding procedure. This would be the same as scratching the BCH codes before Peterson discovered with a

decoding algorithmn for them. Nevertheless, hopeful as we might be, we have to take the decoding problem into consideration when comparing the performance of the coset technique with the other two techniques discussed. However the coset technique is easily encoded once we have the appropriate coset leader C(X). Furthermore it does not alter the length of the parent code V.

In section 3.3. we discussed the Bose-Caldwell technique of dealing with slip errors in the presence of additive errors. The idea there is to construct a special cyclic subcode V* of the parent code V. Each code word of V* has an extra root which is not a root of the words of V which do not belong to V*. This extra root γ is used to generate a slip syndrome from which the slip can be computed. The order n^1 of γ has to be larger than S_L+S_R where S_L is the maximum slip to the left, and S_R is the maximum slip to the right. The words of V have to be lengthened by S_L+S_R symbols and a fixed vector C (X) is added to them before transmission. In other words if the words of V have length n, the words transmitted have length $n+S_L+S_R$. The dimension of V* is $k-m^1$ where k is the dimension of V and m^1 is the degree of the minimal function of γ. Hence when designing

the subcode V *we must try to pick an element γ which has a minimal function of least degree in order to minimize the redundancy introduced. In cases where the order of the Galois Field is a prime all elements have the same order. In such situations we do not have much flexibility in the choice of γ. In such cases we might end up with an inefficient procedure. When comparing the performance of different synchronizable-error-correcting codes, the comparision is only meaningful when comparing codes of the same length. This is so since the probability of having e random errors in $n^1>n$ symbols is most probably larger than the probability of having e errors in n symbols. For example when describing the Bose-Caldwell technique in section 4.3. we assumed that no more that $t=(d-1)/2$ errors occur in a length of $n+S_L+S_R$. Hence when comparing the Bose-Caldwell technique with the coset technique and with the method developed in Chapter 4 we must use an equivalent additive error. To be more specific we will consider e errors in a length $n^1>n$ to be equivalent to $\frac{n}{n^1}e$ errors in a length n. Obviously this is awkward since $\frac{n}{n^1}e$ is not generally an integer. Nevertheless this fact should be kept in mind. The Bose-Caldwell

technique is designed to be used with BCH codes. It is of course not limited to binary codes. This last remark however does not carry that much weight since in most practical systems we are dealing with binary digits more than anything else. The feature which makes the Bose-Caldwell technique attractive is that it has a rather straightforward decoding procedure. Another point to mention is that the amount of left slip need not be equal to the amount of right slip. This feature might be useful in systems with a preferred direction of slip.

In Chapter 4 we described a new approach to the problem of synchronization. Firstly we developed a series of Theorems for the Noiseless case. For this particular situation it was seen that Fire codes are more efficient. The theorem which illustrates the basic philosophy of this chapter is Theorem 4.2.2. For this theorem we presupposed the existence of a binary cyclic code with the property that it can correct the simultaneous occurence of two bursts each of length s^1 or less. This new approach to the problem raises the question as to how to construct such codes. This problem was not dealt with in this thesis. However it is hoped that someone might investigate this problem.

It was then shown that a slight modification of the decoding procedure of Theorem 4.2.2. allows us to use a parent code V which need only correct a single burst of length $3S^1$.

The central theorem of Chapter 4 is Theorem 4.2.5. In this theorem we presuppose the existence of a binary cyclic code which can correct two bursts each of length S^1 or less and an additive error $E(X)$. In any case a t-random error correcting code can always be used instead where $E(X)$ is a random additive error of weight e or less. This particular case is covered in Corallory 4.2.4. However if the amount of slip S^1 is small compared with the additive error e then random error correcting codes become efficient. Since in this case we shortened the code to $n^1=n-2S^1$, we also have to consider an equivalent additive error as in the case of BCH codes. A slight modification of theorem 4.2.5. is presented as Theorem 4.2.7. In this latter situation the length of the code is not altered. This allows us to use a parent code V which need only be capable of correcting 1 burst of length S or less and an additive error $E(X)$. However the transmission rate of this latter code is less than the code of Theorem 4.2.5.

Nevertheless when applied to random error correcting codes this last Theorem does in most cases lead to a more efficient code (see table 5.1.3.). The beauty about this technique is that it has an extremely easy decoding procedure. Furthermore it can be applied to any binary cyclic code with the prescribed properties. If we are forced to use random error correcting codes then we need not use BCH codes. Any efficient random error correcting binary cyclic code will do.

It is difficult to compare the performance of the three techniques described. Many factors have to be kept in mind such as ease of decodability, flexibility in the choice of e and s and finally the transmission rate or the efficiency of the code chosen. Nevertheless we give tables of results which may be used to evaluate the performance of the different techniques.

1	2	3	4	5	6	7	8	9	10	11	12	13
				Theo. 3.2.3			Cor. 4. 2. 4			Cor. 4 2. 5		
(n,k,t)	d	k/n	e	(n,k)	s	k/n	(n',k')	s	k'/n'	(n,k')	s	k'/n'
(127,85,6)	13	.670	1	(127,85)	3	.670	(123,80)	2	.650	(127,74)	5	.582
			2	"	1	"	(123,80)	2	.650	(127,76)	4	.598
			3	"	0	"	(125,82)	1	.655	(127,78)	3	.614
			4				(125,82)	1	.655	(127,80)	2	.630
			5				(127,85)	0	.670	(127,82)	1	.645
			6							(127,85)	0	.670
(255,191,8)	17	.750	1	(255,191)	5	.750	(249,184)	3	.740	(255,176)	7	.690
			2	"	3	"	(249,184)	3	.740	(255,178)	6	.698
			3	"	1	"	(251,186)	2	.742	(255,180)	5	.706
			4	"	0	"	(251,186)	2	.742	(255,182)	4	.713
			5				(253,188)	1	.744	(255,184)	3	.722
			6				(253,188)	1	.744	(255,186)	2	.730
			7				(255,191)	0	.750	(255,188)	1	.738
			8							(255,191)	0	.750
(255,163,12)	25	.640	1	(255,163)	9	.640	(245,152)	5	.62	(255,140)	11	.550
			2	"	7		(245,152)	5	.62	(255,142)	10	.557
			3	"	5		(247,154)	4	.623	(255,144)	9	.565
			4	"	3		(247,154)	4	.623	(255,146)	8	.573
			5	"	1		(249,156)	3	.627	(255,148)	7	.580
			.	"	0		.	.		.	.	
			.				.	.		.	.	
			10				(253,160)	1	.633	(255,158)	2	.620
			11				(255,163)	0	.640	(255,160)	1	.628
			12							(255,163)	0	.640
(127,22,23)	47	.173	1	(127,22)	20	.173	-	.		-		
			2		18	"	(107,1)	10	.00935	-		
			3		15	"	(107,1)	10	.00935			
			.			"	.					
			.			.	.					
			9		4	"	(113,7)	7	.062	-		
			.	-	-	-	.					
			.				.					
			21				(125,19)	1	.149	(127,17)	2	.134
			22					0		(127,19)	1	.150
			23					0		(127,22)	0	.173

TABLE 5.1

Legend to Table 5.1

Column 1.	(n,k,t) of the parent code
Column 2.	d is the minimum distance of the parent code.
Column 3.	k/n is the transmission rate of the parent code.
Column 4.	e is the additive error correcting capacity of the synchronizable codes.
Columns 5,6 and 7.	Refer to Theorem 3.2.3 due to Tavares and Fukada. s is the magnitude of the slip.
Columns 8,9 and 10.	Refer to Corollary 4.2.4.
Columns 11,12,13.	Refer to Corollary 4.2.5.

1	2	3	4	5	6	7	8	9	10	11
		Cor. 4.2.4			Cor. 4.2.5			Bose-Caldwell		
e	s	(n, k, t)	(n', k')	$\mathfrak{R}(V')$	(n, k, t)	(n, k')	$\mathfrak{R}(V_o')$	(n, k, t)	(n*, k*)	$\mathfrak{R}(V^*)$
1	1	(15, 5, 3)	(13, 2)	.154	(15, 7, 2)	(15, 4)	.267	(15, 11, 1)	(17, 9)	.530
1	1	(31, 16, 3)	(29, 13)	.450	(31, 21, 2)	(31, 18)	.580	(31, 26, 1)	(33, 24)	.727
3	1	(31, 11, 5)	(29, 8)	.276	(31, 11, 5)	(31, 8)	.258	(31, 16, 3)	(33, 14)	.425
1	1	(63, 45, 3)	(61, 42)	.690	(63, 51, 2)	(63, 48)	.740	(63, 57, 1)	(65, 55)	.847
3	1	(63, 36, 5)	(61, 33)	.542	(63, 39, 4)	(63, 36)	.572	(63, 45, 3)	(65, 42)	.647
1	2	(63, 36, 5)	(59, 31)	.525	(63, 45, 3)	(63, 40)	.636	(63, 57, 1)	(67, 54)	.806
4	1	(63, 30, 6)	(61, 27)	.443	(63, 36, 5)	(63, 33)	.523	(63, 39, 4)	(65, 37)	.570
4	1	(127, 85, 6)	(125, 82)	.655	(127, 92, 5)	(127, 89)	.700	(127, 99, 4)	(129, 92)	.713
3	2	(127, 78, 7)	(123, 73)	.594	(127, 92, 5)	(127, 87)	.685	(127, 106, 3)	(131, 99)	.755
7	1	(127, 71, 9)	(125, 68)	.545				(127, 78, 7)	(129, 71)	.550
11	1	(127, 50, 13)	(125, 47)	.376				(127, 57, 11)	(129, 50)	.387
4	1	(255, 207, 6)	(253, 204)	.807	(255, 215, 5)	(255, 212)	.830	(255, 223, 4)	(257, 221)	.860
12	1	(255, 147, 14)	(253, 144)	.570	(255, 155, 13)	(255, 152)	.596	(255, 163, 12)	(257, 161)	.626
9	3	(255, 139, 15)	(249, 132)	.530	(255, 163, 12)	(255, 156)	.612	(255, 187, 9)	(261, 183)	.702

TABLE 5.2

Legend to Table 5.2

Column 1. **e** is the number of additive errors.

Column 2. **s** is the magnitude of the slip.

Column 3. **(n,k,t)** of the parent code.

Column 4. **(n', k')** of the modified code.

Column 5. **k'/n'** is the transmission rate of the synchronizable code.

All other columns are self explanatory.

REFERENCES

[1] W.W. Peterson, "Error Correcting Codes", M.I.T. Press, Cambridge Mass., 1961.

[2] E.R. Berlekamp, "Algebraic Coding Theory", McGraw-Hill, New York, 1968.

[3] R.B. Ash, "Information Theory", Interscience Publishers, New York, 1965.

[4] R.W. Hamming, "Error Detecting and Correcting Codes", Bell System Tech. J., Vol. 29, pp. 147-160, 1950.

[5] I.N. Herstein, "Topics in Algebra", Blaisdell Publ. Co., New York, 1964.

[6] C.E. Shannon and W. Weaver, "Mathematical Theory of Communication", University of Illinois Press, Urbana, 1949.

[7] I.S. Reed, "A Class of Multiple-Error-Correcting Codes and the Decoding Scheme", IRE Trans., PGIT-4, pp 38-49, 1954.

[8] S.W. Golomb, "Shift Register Sequences", Holden-Day Inc., San-Francisco, 1967.

[9] A. Hocquenghen, "Codes Correcteurs d'Erreurs", Chiffres, Vol. 2, pp. 147-156, 1959.

[10] R.C. Bose and D.K. Ray-Chaudhuri, " On a Class of Error Correcting Binary Group Codes", Inf. and Control, Vol. 3, pp. 68-77, 1960.

[11] D. Gorenstein, W.W. Peterson, and N. Zierler, "Two Error Correcting Bose-Chaudhuri Codes are Quasi Perfect' Inf. and Control, Vol. 3, pp. 291-294, 1960.

[12] N. Zierler, "Linear Recurring Sequences", J. Soc. Indust. Appl. Math., Vol. 7, pp. 31-48, 1959.

[13] W.W. Peterson, "Encoding and Error-Correction Procedures for the Bose-Chaudhuri Codes", IEEE Trans. on Inf. Theo., Vol. 6, pp. 459-470, 1960.

[14] I.T. Adamson, "Introduction to Field Theory", Oliver and Boyd, Edinburgh, 1964.

[15] E. Artin, "Galois Theory", Notre Name Mathematical Lectures No. 2, 1948.

[16] B.L. van der Waerden, "Modern Algebra", Frederick Ungar Publishing Co., New York, 1931.

[17] A.A. Albert, "Fundamental Concepts of Higher Algebra", University of Chicago Press, Chicago, Ill., 1956.

[18] P. Fire, "A Class of Multiple-Error-Correcting Binary Codes for Non-Independent Errors", Sylvania Report RSL-E-2, Sylvania Reconnaissance Systems Laboratory, Mountain View, Calif., 1959.

[19] S.W. Golomb et al., "Synchronization", IEEE Trans. on Comm. Systems, Vol. 11, pp. 481-492, 1963.

[20] R.H. Barker, "Group Synchronization of Binary Digital Systems, Communications Theory, W. Jackson, Ed., London, England, pp. 273-287, 1953.

[21] J.J. Stiffler, "Commo-Free Error-Correcting Codes", IEEE Trans. on Inf. Theo., Vol. 11, pp. 107-112, 1965.

[22] S.Y. Tong, "Synchronization Recovery Techniques for Binary Cyclic Codes", Bell System Technical J., Vol. 45, pp. 561-596, 1966.

[23] S.E. Tavares, and M. Fukada, "Matrix Approach to Synchronization Recovery for Binary Cyclic Codes", IEEE Trans on Inf. Theo., Vol. 15, pp. 93-101, 1969.

[24] E.N. Gilbert, "Synchronization of Binary Messages", IEEE Trans. on Inf. Theo., Vol. 6, pp. 470-477.

[25] S.W. Golomb, B. Gordon, and L.P. Welch, "Comma-Free Codes", Canadian Journal of Mathematics, Vol. 10, pp. 202-209, 1958.

[26] R.C. Bose and J.G. Caldwell, "Synchronizable Error-Correcting Codes", Inf. and Control, Vol. 10, pp. 616-630, 1967.

[27] S.E. Tavares, "A Study of Synchonization Techniques for Binary Cyclic Codes", Ph.D. Thesis, Dept. of Elect. Engrg., McGill University, Montreal, Canada, 1968.

[28] J.G. Caldwell, "Synchronizable Error-Correcting Codes", Doctoral Disertation, Department of Statistics, Univ. of North Carolina, Chapel Hill, 1966.

[29] - Same as 25.

[30] J. M. Wozencraft and I.M. Jacobs, "Principles of Communication Engineering", John Wiley and Sons Inc., New York, 1967.

[31] H. Paley and P.M. Weichsel, "A First Course in Abstract Algebra", Holt, Rinehart and Winston Inc., New York, 1966.

[32] S.G.S. Shiva and T. Zeitoun, "Shortened Cyclic Burst Error Correcting Binary Codes of Certain Length", Tech. Report No. 68-5 Dept. of Elect. Engrg., University of Ottawa, Canada, 1968.

[33] J. J. Stone, "Multiple Burst Error Correction", Inf. and Control, Vol. 4, pp. 324-331, 1961.

[34] J.J. Stone, "Multiple-Burst Error Correction with the Chinese Remainder Theorem", J. Soc. Indust. Appl. Math., Vol 11, pp. 74-81, 1963.

[35] E.J. Weldon Jr., "A Note on Synchronization Recovery with Extended Cyclic Codes", Inf. and Control, Vol. 13, pp. 354-356, 1968.

[36] S.E. Tavares and M. Fukada, "Synchronization of Cyclic Codes in the Pressence of Burst Errors", Inf. and Control, Vol. 14, 1969, pp. 423-441.

[37] P. Mandelbaum, "A Note on Synchronizable Error Correcting Codes", Information and Control, Vol. 13, 1968, pp. 429-432.

[38] S.G.S. Shiva and G. Seguin, "On the Correction of Synchronization and Additive Errors", presented at the 1969 International Symposium on Information Theory, Ellenville, N.Y. January 1969. Also accepted for publication in IEEE Trans. on Inf. Theo. .

[39] J.E. Levy, "Self Synchronizing Codes Derived From Binary Cyclic Codes", IEEE Trans. on Inf. Theo., Vol. IT-12 No. 3, pp. 286-290, July 1966.

VITAE

NAME: Gerald Seguin

BORN: Alexandra, Ontario.

SCHOOL:

Secondary: Eastview High School.

University: University of Ottawa, B. Sc. 1967