SELF-SYNCHRONIZING BINARY
TELEMETRY CODES

Thesis by

Jack J. Stiffler

In Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy

California Institute of Technology

Pasadena, California

1962

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor, Dr. S. W. Golomb, for his very capable direction and his many helpful suggestions.

I also wish to acknowledge the financial support provided by the Ford Foundation and by the Jet Propulsion Laboratory during the period over which this thesis was written.

ABSTRACT

The past decade has witnessed significant advances in the techniques for communication, with high reliability, over noisy channels and, in particular, in the methods of encoding for these channels. However, for many of these encodings, including the so-called block codes, efficient reception demands that the receiver know the instant in time that one block of data ends and the succeeding block begins. This synchronization problem, as applied to an important class of block codes, which are optimum or nearly optimum over the continuous white Gaussian channel, is the central topic treated in this thesis. Two synchronization methods are presented, and upper bounds on the time necessary for their operation are determined. The first involves almost no additional encoding or decoding equipment, but is dependent upon the randomness of the received message. The second technique, while necessitating more complex decoding apparatus, is, in general, considerably more rapid than the first and is, moreover, independent of the statistical properties of the data. Neither method decreases the information capacity of the channel. The performance of both these techniques in conjunction with the binary symmetric channel is also investigated.

TABLE OF CONTENTS

Part	Title	Page
Introduction		1
Chapter 1	ORTHOGONAL, BI-ORTHOGONAL AND TRANS-ORTHOGONAL CODES	
A.	Synchronous Word Error Probabilities	7
B.	Bit Error Probabilities for Orthogonal Codes	15
C.	Bandwidth Occupancy for Orthogonal Codes	16
D.	Optimum Coding for the Continuous, Noisy Channel	16
E.	Binary Codes	19
F.	Bi-Orthogonal and Trans-Orthogonal Word Error Probabilities	24
G.	Bandwidth Occupancy for Binary Orthogonal Codes	25
H.	Some Comments on Mechanization	29
I.	Symbol Synchronization	32
Chapter 2	SYNCHRONIZATION USING THE PROPERTIES OF A RANDOM SOURCE	
A.	The Probability of Correct Synchronization	33
B.	Asymptotic Results	39
C.	An Alternate Synchronization Technique	40
Chapter 3	COMMA-FREE CODES	
A.	Comma Freedom and Its Importance	42
B.	The Number of Distinct Orthogonal Dictionaries of N Words	47
C.	The Structure of Orthogonal Groups	57
Chapter 4	BOUNDS ON THE INDEX OF COMMA FREEDOM FOR ORTHOGONAL CODES	
A.	Upper Bounds	60
B.	Lower Bounds	67

TABLE OF CONTENTS (continued)

Part	Title	Page
Chapter 5	SYNCHRONIZATION WITH COMMA-FREE CODES	
A.	The Probability of Correct Synchronization	75
B.	Asymptotic Results	85
C.	Some Comments Concerning the Complexity of the Decoder	87
Chapter 6	SYNCHRONIZATION OVER THE BINARY SYMMETRIC CHANNEL	
A.	The Binary Symmetric Channel	89
B.	Word Synchronization by the Random Source Method	96
C.	Word Synchronization with Comma-Free Codes	99
Chapter 7	A METHOD OF ANALYZING THE COMMA-FREE PROPERTIES OF GROUP CODES	
A.	The Matrix Equation	110
B.	The Non-Existence of Orthogonal Comma-Free Group Codes of Eight Words	117
Chapter 8	SUPPLEMENTARY REMARKS CONCERNING SYNCHRONIZATION	
A.	Other Approaches to Synchronization	122
B.	Frame Synchronization	125
C.	Retention of Word Synchronization	127
Appendix A	EXAMPLES OF COMMA-FREE DICTIONARIES	129
Appendix B	SOME PARTICULAR UPPER BOUNDS ON THE INDEX OF COMMA FREEDOM USING THE MATRIX EQUATION	139
References		148

INTRODUCTION

A current problem which is receiving considerable attention under the impetus of satellite and deep-space communication requirements is concerned with the transmission of information through the relatively unconstrained continuous channel[1]. The communication channel model of Figure 1 may be used to describe the conditions encountered in this situation.

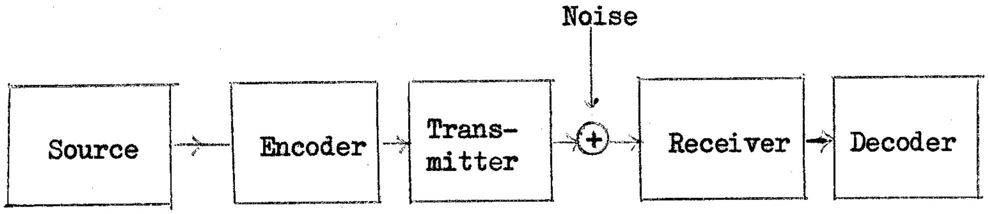


Figure 1

The source consists of, for example, the data from numerous experiments conducted during flight. The encoder converts this source message into a more convenient form for transmission, presumably with the intention of increasing the reliability of the system, while the decoder reverses the process. The noise is primarily thermal in origin, and its power is constant over a very wide bandwidth. Fading and multi-path difficulties are negligible. The common assumption that the noise is stationary, white, and Gaussian would seem to be well-founded in this case, and experimental evidence provides excellent verification of this conjecture.

The encoder is assumed to be a block encoder; that is, the data is quantized and represented by a sequence of N-ary numbers and each of these N possible data digits is represented by a particular waveform or code word for transmission. The collection of these code words is referred to as the code dictionary. It will also be assumed that each of the N data digits has equal probability of occurring and that the words used to transmit these digits have equal time duration, T, and equal energy, E_o.

A continuous sequence of words selected by the source from these N words is then used to modulate a carrier which is in turn transmitted through the channel. After demodulation the decoder estimates which of the N possible words is represented by each successive T seconds of the incoming sequence. More precisely, the decoder attempts to determine which of the N waveforms, $x_i(t)$ $(i = 1, 2, \ldots, N)$, was most likely transmitted after having received a particular waveform, $y(t)$, which has been corrupted by additive noise. This most probable transmitted code word is determined by the value of $i = 1, 2, \ldots, N$ for which the probability $P(x_i|y)$ is maximized. But by Bayes' rule

$$P(x_i|y)\ p(y)\ dy = p(y|x_i)\ P(x_i)\ dy$$

and

$$\max_i P(x_i|y) = \max_i \frac{p(y/x_i)\ P(x_i)}{p(y)} \quad . \tag{0.1}$$

Thus it is sufficient to find the value of i which maximizes $p(y/x_i)\ P(x_i)$. This decision criterion is known as the maximum a posteriori criterion. The situation of interest here, in which the a priori probabilities $P(x_i)$ are either unknown, or are assumed to be equal, results in the so-called maximum likelihood criterion in which the value of i maximizing $p(y/x_i)$ is to be determined. Optimum detectors are known for both these cases(2,3,4,5) when the noise is white and Gaussian. If the energy of the transmitted signal, $E_o = \int_o^T x_i^2(t)\, dt$, is the same for all waveforms, x_i, the optimum detector according to the maximum likelihood criterion is one which forms the integral,

$$\phi_i(t_o) = \int_{-T}^{o} x_i(-\tau)\, y(t_o - \tau)\, d\tau, \qquad (0.2)$$

and selects that x_i corresponding to $\max_{i,t_o} \phi_i(t_o)$, $i = 1, 2, \ldots, N, 0 \leq t_o < T$ as the received code word. $\phi_i(t_o)$ is thus the response to the signal $y(t_o - \tau)$ of a filter whose impulse response is $x_i(-\tau)$, the well-known matched-filter. Alternatively, setting $t_o - \tau = t$ in the above expression for $\phi_i(t_o)$, one obtains:

$$\phi_i(t_o) = \int_{t_o}^{t_o + T} x_i(t - t_o)\, y(t)\, dt. \qquad (0.3)$$

But this is the correlation between $x_i(t - t_o)$ and $y(t)$ and may be readily mechanized by the following correlation detector:

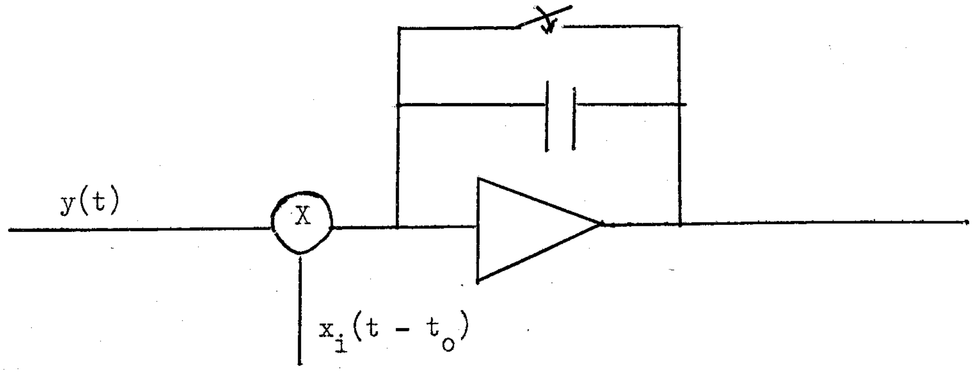


Figure 2

where the switch is opened at time t_o and the output observed at time $t_o + T$. The relative amplitude of the local waveform, $x_i(t - t_o)$, is, of course, irrelevant and will be assumed, for convenience, to be unity.

Note that the maximum likelihood criterion states that the waveform most likely transmitted and the time interval over which it occurred is given by $\max\limits_{\substack{i = 1, 2, \ldots, N \\ 0 \leq t_o < T}} \phi_i(t_o)$. Thus there are N competing waveforms and a continuum of competing time instants t_o. A better estimate of the complexity of the system necessary to determine the most likely code word is obtained by noting that the number of independent samples per second that can be transmitted through a channel of bandwidth W is approximately 2W.(6) Thus there are 2WT independent parameters defining each word and N words, or a total complexity proportional to 2WNT. Further, it is very possible that some of the waveforms, y(t), of length T, which are not actually code words but formed by the combination of two code words, are nearly, or even exactly, equivalent to true code words. The occurrence of only one such sequence alone reduces the probability of identifying the true code word to approximately $\frac{1}{2}$. In the application under consideration here, however, a continuous sequence of code words is being sent. The instant one word ceases, another begins. In addition, it will be assumed that time T is known exactly (i.e., that a common transmitter-receiver time reference exists, as provided, for example, by the coherent reception of the carrier or subcarrier). If the value of t_o is known for one code word in the sequence, it is thus known for the duration of the sequence, and the number of "competitors" is then reduced from 2WNT to just N for each code word received. The probability of an error is thus significantly decreased and the mechanization process vastly simplified by a good estimate of the time t_o.

It is the purpose of this paper to establish ways of determining, as rapidly and accurately as possible, the value of t_o (that is, to obtain word synchronization) for a particular class of block codes.

It should be acknowledged that an obvious alternative here is to incorporate a second channel in the system which sends nothing but synchronization information. As soon as synchronization is obtained, however, the power relegated to this channel is entirely useless and could presumably be used to transmit information. Two-way communication could overcome this difficulty by switching off the synchronization channel when it is no longer needed. For space probes, however, the round trip communication time is appreciable and could result in many minutes of wasted power. These comments are equally applicable to the single channel system in which certain patterns which are to be used for synchronization are periodically transmitted[(7)] since this, too, decreases the information capacity of the channel. Certainly, the advantages to be gained by having "self-synchronizing" codes in decreasing the amount and complexity of equipment carried on the probe, as well as in the conservation of power, are not insignificant, if such codes exist. In order not to diminish these advantages, however, several conditions may be imposed upon the self-synchronizing code: (1) The expected, or maximum, time necessary to obtain synchronization must be small (if real time decoding is to be accomplished) or at least the synchronization time must be no more than the minimum continuous time interval over which the code is to be received (if non-real time decoding is all that is demanded), (2) The error probability in synchronous operation must not be increased with a resulting decrease in the information rate due to the addition of self-synchronizing properties, and (3) No redundancy should be introduced, thereby reducing the channel capacity, for purposes of synchronization.

It might appear at first sight that from an information theoretic point of view, the task outlined could not be accomplished under the conditions listed. That is, each correctly detected word conveys exactly $n = \log_2 N$ bits of information, and since no redundancy is to be added to facilitate the synchronization process, these n bits must correspond to n bits of data information. Thus there are no surplus bits available to carry the synchronization information. However, it is also true that the asynchronous error probability is greater and that consequently the information rate(8) is less than that possible with the same channel at the same signal-to-noise ratio after synchronization has been obtained. This difference between the asynchronous and synchronous information rate represents a rate at which it is theoretically possible to send synchronization information without violating any of the basic principles of information theory. The following chapters are an attempt to utilize some of this synchronization capacity.

Chapter 1

ORTHOGONAL, BI-ORTHOGONAL AND TRANS-ORTHOGONAL CODES

A. Synchronous Word Error Probabilities

Consider now the probability of correctly determining which of the N signals, s_1, s_2, ..., s_N, has actually been sent assuming now that word synchronization has been established. The maximum likelihood detector then forms the integral

$$Z_{ik} = \int_0^T y(t)\, s_i(t)\, dt = \int_0^T \left[As_k(t) + n(t)\right] s_i(t)\, dt \qquad (1.1)$$

where A represents the amplitude of the received signal relative to that of the transmitted signal. Since $s_i(t)$ is a deterministic signal, and n(t) is produced by a Gaussian random process, Z_{ij} is a Gaussian random variable.(9) The joint probability density function,

$$p(Z_{1k}, Z_{2k}, \ldots, Z_{Nk}) = \frac{|\Lambda|^{\frac{1}{2}}}{(2\pi)^{N/2}} \; e^{-(Z - m)\Lambda(Z - m)^T} \qquad (1.2)$$

where Z is the row vector with components Z_{1k}, Z_{2k}, ..., Z_{Nk}, m the vector whose components are the expected values of the components of Z, m_{1k}, m_{2k}, ..., m_{Nk}, Λ, the inverse of the covariance matrix, and $|\Lambda|$ the determinant of Λ. The components of m are determined by

$$m_{ik} = E(Z_{ik}) = \int_0^T As_i(t)\, s_k(t)\, dt \qquad (1.3)$$

since $E\left[n(t)\right] = 0$. Similarly, the elements of the matrix Λ^{-1}, the covariance matrix, are given by $\lambda_{ij}^{-1} = \sigma_i(k)\,\sigma_j(k)\,\rho_{ij}(k)$ where

$$\sigma_i(k)\,\sigma_j(k)\,\rho_{ij}(k) = E\Big[(Z_{ik} - m_{ik})\,(Z_{jk} - m_{jk})\Big]$$

$$= E \int_o^T \int_o^T \Big[As_k(t) + n(t)\Big]\Big[As_k(u) + n(u)\Big]\, s_i(t)\, s_j(u)\, dtdu - m_{ik}\, m_{jk}$$

$$= \sigma_n^2 \int_o^T s_i(t)\, s_j(t)\, dt \qquad (1.4)$$

since $E\Big[n(t)\,n(u)\Big] = \sigma_n^2\,\delta(t - u)$ for white noise with $2\,\sigma_n^2 = N_o$, the noise power spectral density. Finally,

$$\sigma_i^2(k) = E\Big[(Z_{ik} - m_{ik})^2\Big] = \sigma_n^2 \int_o^T s_i^2(t)\, dt. \qquad (1.5)$$

Note that both $\sigma_i(k)$ and $\rho_{ij}(k)$ are independent of the actual word s_k that was transmitted.

In accordance with the maximum likelihood criterion, the largest of the random variables Z_{ik} corresponds to the most probable signal transmitted. The probability that a correct decision is made, given that s_k was transmitted is then:

$$P_c(s_k) = \int_{-\infty}^{\infty} \int_{-\infty}^{Z_{kk}} \int_{-\infty}^{Z_{kk}} \cdots \int_{-\infty}^{Z_{kk}} p(Z_{1k}, Z_{2k}, \ldots, Z_{Nk})\, dZ_{Nk}\, dZ_{N-1,k}\, dZ_{ik}\, dZ_{kk}. \qquad (1.6)$$

The total probability of a correct decision is

$$P_c = \sum_{k=1}^{N} P_c(s_k)\, P(s_k) = \frac{1}{N} \sum_{k=1}^{N} P_c(s_k) \qquad (1.7)$$

under the assumption that the a priori probability that s_k was sent, $P(s_k)$, is independent of k.

Consider now the special case when

$$\int_0^T s_i(t)\, s_j(t)\, dt = E_o\, \delta_{ij} = \begin{cases} E_o & i = j \\ 0 & i \neq j \end{cases} \qquad (1.8)$$

where $E_o = \int_0^T s_j^2(t)\, dt$ is the energy of the jth code word and is assumed to be independent of j. Then,

$$\rho_{ij} = \frac{\int_0^T s_i(t)\, s_j(t)\, dt}{\left(\int_0^T s_i^2(t)\, dt \int_0^T s_j^2(t)\, dt \right)^{\frac{1}{2}}} = \delta_{ij}$$

$$m_{ik} = A\, E_o\, \delta_{ik} \qquad (1.9)$$

$$\sigma^2 = \sigma_n^2\, E_o$$

and

$$\Lambda = (\Lambda^{-1})^{-1} = \frac{1}{\sigma_n^2\, E_o}\, I$$

where I is the identity matrix. A code dictionary with this property is known as an orthogonal code. The error probability becomes simply:

$$P_c = \frac{1}{N} \sum_{k=1}^{N} P_c(s_k) = P_c(s_k) =$$

$$\frac{1}{(2\pi)^{N/2} \sigma_n^{N} E_o^{N/2}} \int_{-\infty}^{\infty} \exp\left\{ - \frac{(Z_k - AE_o)^2}{2\sigma_n^2 E_o} \right\} \left[\int_{-\infty}^{Z_K} \exp\left\{ - \frac{Z_o^2}{2\sigma_n^2 E_o} \right\} dZ_o \right]^{N-1} dZ_k \qquad (1.10)$$

$$= \frac{1}{(2\pi)^{N/2}} \int_{-\infty}^{\infty} e^{-\frac{s^2}{2}} \left[\int_{-\infty}^{s + \frac{AE_o^{\frac{1}{2}}}{\sigma_n}} e^{-\frac{t^2}{2}} dt \right]^{N-1} ds.$$

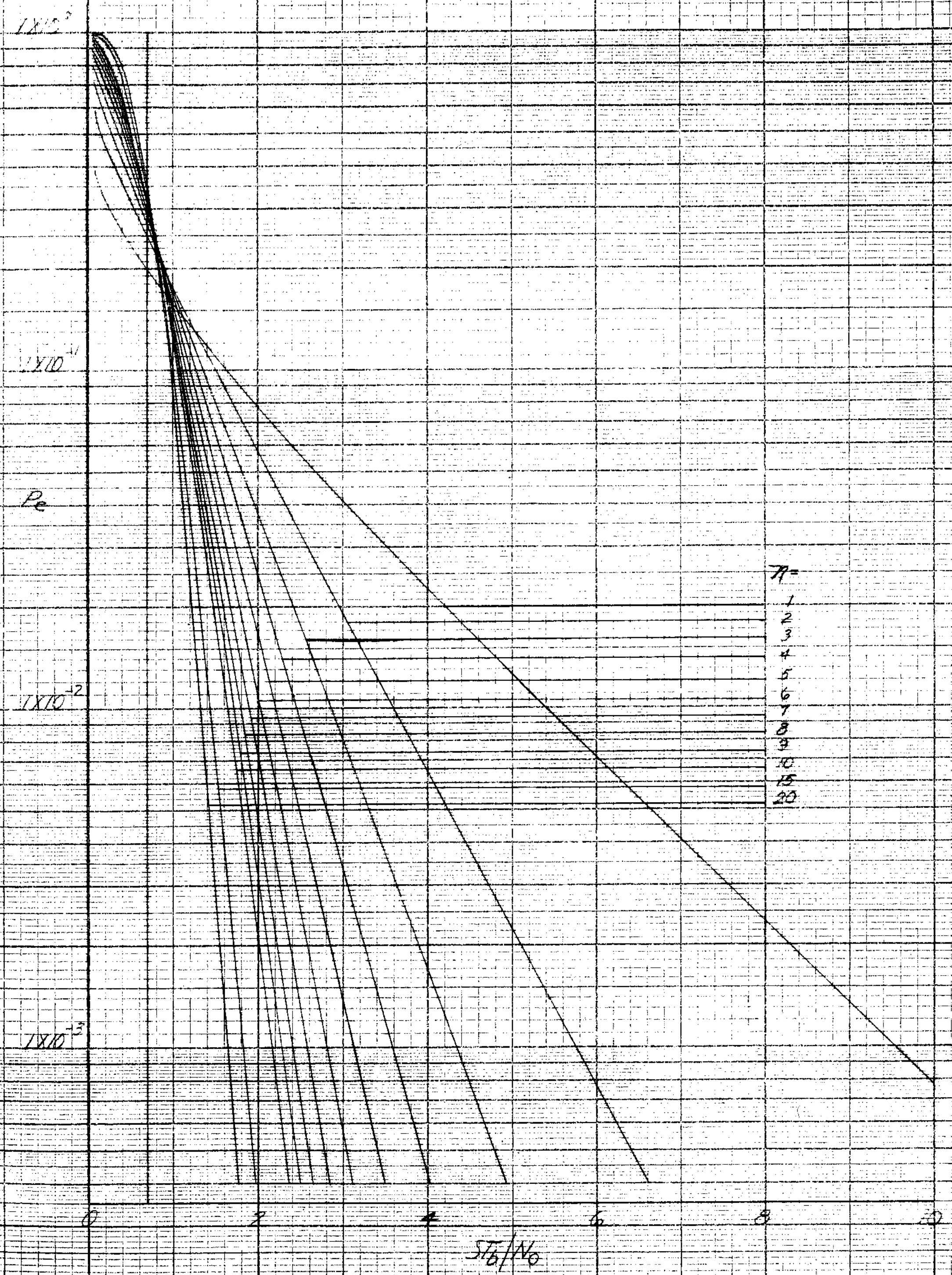
The value of $P_e = 1 - P_c$, the probability of a word error for orthogonal codes, has been calculated numerically(10) and is plotted for a fixed $n = \log_2 N$ in Figure 1.1, as a function of $\frac{1}{2} \frac{A^2 E_o}{n} \frac{1}{\sigma_n^2} = \frac{ST_b}{N_o}$, the ratio of the received signal energy per information bit* to the noise power per unit bandwidth. S here denotes the average received signal power, $T_b = T/n$ is the time per bit.

An upper bound for P_c can be obtained for the orthogonal case by observing that in the expression

* It is apparent that the signal energy per information bit, not the signal energy per word, is the pertinent parameter since longer words, when correctly detected, convey more information. The number of bits of information corresponding to one of N equally probable words is, of course, $n = \log_2 N$.

FIGURE 1.1

ORTHOGONAL CODES:
WORD ERROR PROBABILITY



$$P_c = \int_{-\infty}^{\infty} p(x) \left[P(y< x)\right]^{N-1} dx,$$

the term

$$\left[P(y < x)\right]^{N-1} = \left[1 - P(y > x)\right]^{N-1} \geq 1 - (N-1)\ P(y > x).$$

Thus

$$P_c \geq 1 - (N-1) \int_{-\infty}^{\infty} p(x)\ P(y > x)\ dx$$

$$= 1 - \frac{N-1}{2\pi} \int_{-\infty}^{\infty} \int_{x + \frac{\mu_x}{\sigma}}^{\infty} e^{-\frac{x^2}{2}}\ e^{-\frac{y^2}{2}}\ dydx\ . \qquad (1.11)$$

Now letting $s = \frac{1}{\sqrt{2}}(x + y)$, $t = \frac{1}{\sqrt{2}}(x - y)$ and observing that the Jacobian(11) $\left|J\right| = 1$, it is seen that

$$P_c \geq 1 - \frac{N-1}{2\pi} \int_{-\infty}^{\infty} \int_{-\frac{\mu_x}{\sqrt{2}\sigma}}^{-\infty} e^{-\frac{(s+t)^2}{4}}\ e^{-\frac{(s-t)^2}{4}}\ dtds$$

$$= 1 - \frac{N-1}{2\pi} \int_{-\infty}^{\infty} e^{-\frac{s^2}{2}}\ ds \int_{\frac{\mu_x}{\sqrt{2}\sigma}}^{\infty} e^{-\frac{t^2}{2}}\ dt \qquad (1.12)$$

$$= 1 - (N-1) \left[1 - \operatorname{erf}\left(\frac{\mu_x}{\sqrt{2}\,\sigma} \right) \right] .$$

Since, in this case, $\mu_x = AE_o$ and $\sigma = \sigma_n E_o^{\frac{1}{2}}$, the above expression becomes

$$P_c \geq 1 - \frac{(N-1)}{(2\pi)^{\frac{1}{2}}} \int_{\frac{AE_o^{\frac{1}{2}}}{\sqrt{2}\sigma_n}}^{\infty} e^{-\frac{t^2}{2}} dt \geq 1 - \frac{\sigma_n (N-1)}{\pi^{\frac{1}{2}} AE_o^{\frac{1}{2}}} e^{-\frac{A^2 E_o}{4\sigma_n^2}} \qquad (1.13)$$

The last step follows from the well-known inequality

$$\frac{1}{(2\pi)^{\frac{1}{2}}} \int_{\alpha}^{\infty} e^{-\frac{t^2}{2}} dt \leq \frac{1}{(2\pi)^{\frac{1}{2}}\alpha} e^{-\frac{\alpha^2}{2}} \qquad \alpha \geq 0.$$

Substituting $\frac{2n\, ST_b}{N_o}$ for $\frac{A^2 E_o}{\sigma_n^2}$ and observing that $(N-1) = e^{\log_e 2 \log_2 (N-1)}$ $< e^{\log_e 2 \log_e N}$ and that $n^{-\frac{1}{2}} = e^{-\frac{1}{2}\log_e(\log_2 N)}$, one obtains:

$$P_e = 1 - P_c \leq \left[\frac{N_o}{2\pi ST_b} \right]^{\frac{1}{2}} e^{\log_e 2 \log_2 N - \frac{1}{2}\frac{ST_b}{N_o}\log_2 N - \frac{1}{2}\log_e(\log_2 N)} \qquad (1.14)$$

$$\leq \left[\frac{N_o}{2\pi ST_b} \right]^{\frac{1}{2}} e^{-\left[\frac{1}{2}\frac{ST_b}{N_o} - \log_e 2\right] n} .$$

Thus, for $\frac{ST_b}{N_o} > 2\log_e 2$, the upper bound to the error probability approaches zero asymptotically as the number of bits per word. Actually, for the error probability to become zero asymptotically with n, it is only necessary that $\frac{ST_b}{N_o} > \log_e 2$ as the following argument will show. An error occurs only when the random sample x_o from the Gaussian distribution with a mean $\mu = AE_o$ and variance $\sigma^2 = \sigma_n^2 E_o$ (corresponding to the output of the correct correlator) is less than the largest of N-1 independent samples from a Gaussian distribution with the same variance but with zero mean (corresponding to the largest output of the N-1 incorrect correlators). It can be shown(12,13,14) that the distribution of the largest of N-1 samples from a Gaussian distribution with mean μ and variance σ has a mean $\mu_L \sim \mu + (2\log_e(N-1))^{\frac{1}{2}}\sigma$ and a variance $\sigma_L^2 \sim \frac{\sigma^2}{\log_2(N-1)}$, asymptotic with N. Thus, asymptotically, no error will occur if

$$x_o > \left[2\log_e(N-1)\right]^{\frac{1}{2}}\sigma \approx \left[2\log_e N\right]^{\frac{1}{2}}\sigma$$

and

$$P_e \approx \int_{-\infty}^{\left[2\log_e(N)\right]^{\frac{1}{2}}\sigma} \left(\frac{e^{-\frac{(x_o - \mu)^{\frac{1}{2}}}{2\sigma^2}}}{\sqrt{2\pi}\,\sigma} \right) dx_o$$

$$= \int_{-\infty}^{-\left[2\log_2 N\right]^{\frac{1}{2}} \left[\left(\frac{ST_b}{N_o}\right) - (\log_e 2)\right]^{\frac{1}{2}}} \frac{e^{-\frac{t^2}{2}}}{\sqrt{2\pi}}\, dt. \qquad (1.15)$$

The upper limit then is asymptotically $-\infty$ if $\frac{ST_b}{N_o} \geq \log_e 2$ and the error probability thus goes to zero if this inequality holds.

B. Bit Error Probabilities for Orthogonal Codes

It is sometimes preferable to specify the probability that an information bit rather than a word is in error. The former is easily obtained from the latter in the case of orthogonal codes, due to the fact that all N-1 incorrect words are equally likely to be mistaken for the correct word.(15) By Baye's rule, the probability that a bit is in error, P(b), is

$$P(b) = P(b|w)\, P(w)$$

where $P(b|w)$ is the probability that a bit is in error given that a word is in error and $P(w)$ is the word error probability. The expected number of bits in error given that a word is in error is:

$$E(b|w) = n\, P(b|w) = \sum_{i=1}^{n} i\, P(i).$$

$P(i)$, the probability that i bits are in error, is just the ratio of the number of data words which differ from the true word in exactly i bits to the total number of wrong words:

$$P(i) = \frac{\binom{n}{i}}{2^n - 1} .$$

Thus

$$P(b|w) = \frac{1}{n(2^n - 1)} \sum_{i=1}^{n} i\binom{n}{i} = \frac{1}{2^n - 1} \sum_{j=0}^{n-1} \binom{n-1}{j} \qquad (1.16)$$

$$= \frac{2^{n-1}}{2^n - 1} = \tfrac{1}{2}\left(\frac{N}{N-1}\right) .$$

C. Bandwidth Occupancy for Orthogonal Codes

As was mentioned earlier, a sample of duration T seconds of a signal limited to a bandwidth W, can be specified by 2WT real numbers. Any signal may be considered as a vector of 2WT components. An orthogonal code must consist of a subset of these signals with the property that(16)

$$\frac{1}{E_o} \int_o^T s_i(t)\, s_j(t)\, dt = \frac{1}{E_o} \sum_{k=1}^{2WT} a_i(kt_o)\, a_j(kt_o) = \delta_{ij} \qquad (1.17)$$

where $t_o = \frac{1}{2W}$ and $a_i(m)$ is the mth sample of the signal $s_i(t)$ limited to a bandwidth W. That is, the vectors $\left\{ a_i(m) \right\}$ representing the signals $s_i(t)$ must consitute an orthogonal set. It is well-known that such a set is linearly independent and, consequently, can contain only as many members as the dimensionality, 2WT, of these vectors. Thus $N = 2^n \leq 2WT$ and

$$W \geq \frac{2^{n-1}}{T} = \frac{2^{n-1}}{nT_b} \qquad (1.18)$$

D. Optimum Coding for the Continuous, Noisy Channel

It follows from this that if the transmission rate, $R = \frac{1}{T_b}$, is fixed, the bandwidth approaches infinity as n becomes large. Simultaneously, the error probability approaches zero. It is interesting to investigate the capacity of the channel under the same bandwidth conditions. Since for white noise(17):

$$C = W \log_2(1 + \frac{S}{WN_o})$$

then

$$C_{\infty} = \lim_{W \to \infty} W \log_2(1 + \frac{S}{WN_o}) = \lim_{W \to \infty} W \log_e(1 + \frac{S}{WN_o}) \log_2 e \to \frac{S}{N_o} \log_2 e \qquad (1.19)$$

Hence, when

$$\frac{ST_b}{N_o} = \log_e 2$$

$$C = \frac{S}{N_o \log_e 2} = \frac{1}{T_b}$$

and one may transmit information at a rate, $R < \frac{1}{T_b}$, arbitrarily close to the channel capacity with an arbitrarily small probability of error, with an orthogonal code.

Let us denote the integral of equation 1.10 by $\Phi_N(\frac{ST_b}{N_o})$, the probability of correctly determining which of an orthogonal set of N signals of energy E_o was sent in the presence of white Gaussian noise with variance σ_n^2. It has been shown(18,19) that, in the particular case that the ρ_{ij}, $(i \neq j)$, terms of equation 1.4 are all equal to some value, ρ,

(ρ_{ii}, of course, is always equal to one), the value of P_c of equation 1.7 is given by

$$P_c = \Phi_N \left[\frac{ST_b(1 - \rho)}{N_o} \right] . \tag{1.20}$$

In addition, if the ρ_{ij} $(i \neq j)$ are not necessarily equal, but algebraically bounded by some ρ_{max}, then

$$P_c \leq \Phi_N \left[\frac{ST_b(1 - \rho_{max})}{N_o} \right] \tag{1.21}$$

That is, the error probability for the general case is bounded by the error probability obtained in the orthogonal situation when the signal energy is reduced by a factor of $1 - \rho_{max}$. Minimizing this error probability, is, in fact, equivalent to minimizing the value of ρ_{max}.[20]

It is readily seen that[21]

$$\rho_{max} \geq \rho_{ave} = \frac{1}{N(N-1)} \sum_{i \neq j} \rho_{ij} = \frac{1}{N(N-1)} \left(\sum_i \sum_j \rho_{ij} - N \right).$$

From equations 1.4 and 1.5,

$$\rho_{ij} = \sigma_n^{\,2} \int_o^T \frac{s_i(t)}{\sigma_i} \, \frac{s_j(t)}{\sigma_j} \, dt.$$

Thus

$$\rho_{ave} = \frac{1}{N(N-1)} \left\{ \sigma_n^{\,2} \int_o^T \sum_i \frac{s_i(t)}{\sigma_i} \sum_j \frac{s_j(t)}{\sigma_j} \, dt - N \right\} \tag{1.22}$$

$$= \frac{1}{N(N-1)} \left\{ \sigma_n^2 \int_o^T \left(\sum_i \frac{s_i(t)}{\sigma_i} \right)^2 dt - N \right\} \geq - \frac{1}{N-1} \qquad (1.22)$$

(continued)

E. Binary Codes

There are many practical advantages associated with the digital processing of data in a space vehicle. The reliability of digital equipment and the convenience and efficiency of storing digital information are most important assets associated with digital, and more particularly, binary systems. This digital data may be used to select the desired waveform to be transmitted. More conveniently, the data may be used to generate a binary sequence with the desired properties which in turn is used to modulate a subcarrier. The system we will consider here is the following: The binary data is divided into $N = 2^n$ blocks of n digits. Each of these blocks is encoded into a word of M binary digits. The digits of the word which is to be transmitted are then used to phase modulate a subcarrier by 0^o or 180^o depending upon whether the corresponding digit is a zero or a one. Note that this is just a method of implementing the block encoder described in the Introduction.

Let the binary digits comprising the word, hereafter referred to as symbols to distinguish them from the information bits, have a time duration of T_s seconds, and let the subcarrier angular frequency ω_o be some multiple of $\frac{\pi}{2T_s}$. Then since

$$\frac{1}{T_s} \int_0^{T} \cos(\omega_0 t + k\pi) \cos(\omega_0 t + k'\pi)\, dt$$

$$= \frac{1}{T_s} \int_0^{T_s} \cos(2\omega_0 t + (k + k')\pi)dt + \frac{1}{T_s} \int_0^{T_s} \cos(k - k')\pi dt$$

$$= \cos(k - k')\pi$$

the cross-correlation between a one and a zero is -1, $(k-k' = \pm 1)$, while the cross-correlation between a one (or zero) and a one (or zero) is +1, $(k - k' = 0)$. Thus, the normalized correlation between a code word consisting of the binary digits x_i, $(i = 1, 2, \ldots, M)$, and that consisting of the digits y_i, $(i = 1, 2, \ldots, M)$, is given by

$$\rho_{xy} = \frac{1}{M} \sum_{i=1}^{M} (1 - 2x_i)(1 - 2y_i) \qquad (1.23)$$

since the product is one if $x_i = y_i$ and minus one if $x_i \neq y_i$. The substitution, $\xi_i = 1 - 2x_i$, provides an alternate way of representing the code words under consideration and will be used later when convenient. Note also that

$$\rho_{xy} = \frac{A(x, y) - D(x, y)}{M} = 1 - \frac{2D(x, y)}{M} \qquad (1.23a)$$

where $A(x, y)$ is the number of times the corresponding components of the binary vectors x and y are in agreement and $D(x,y)$ is the number of times they disagree.

Suppose now that there are N possible blocks to be transmitted. Is it possible to find N binary vectors of length M such that ρ_{max} (or ρ_{ave}) is constrained to be less than a certain value ρ_o? This is certainly not possible for arbitrary M, N and ρ_o. If two of these three parameters are given, bounds on the third can be derived. For particular values of ρ_o, and these include the most interesting cases, much more complete information is available.

Consider in particular the case in which all correlations are constrained to be zero, the orthogonal case discussed earlier. Considering the code words as vectors of +1's and -1's, they are orthogonal over the field of real numbers. Since these vectors form a basis over this field, there cannot be more than M such vectors, where M is their dimensionality. Thus $N \lesssim M$. Let x, y, and z be three binary code vectors with components x_i, y_i and z_i. Then if these three vectors are to be orthogonal

$$\frac{1}{M} \sum_{i=1}^{M} (1 - 2x_i)(1 - 2y_i) = 0$$

$$\frac{1}{M} \sum_{i=1}^{M} (1 - 2x_i)(1 - 2z_i) = 0$$

and

$$\frac{1}{M} \sum_{i=1}^{M} (1 - 2y_i)(1 - 2z_i) = 0.$$

Combining these three equations, one obtains

$$3M = 4\left(\sum x_i - \sum y_i - \sum z_i + \sum x_iy_i + \sum x_iz_i + \sum y_iz_i\right).$$

Thus 3M is a multiple of four and, if N is greater than 2, M must be a multiple of four. It is then a necessary condition that $N \leq M = 4t$.(22) Solutions have been obtained for $M = N = 4t$ for all t up to 50, except for t = 29, 47 and 49, and for many values of $t > 50$. In addition, if a solution is known for any value of t, it is also known for 2t due to the following construction: Let A be a n x n matrix of ones and minus ones such that the rows are mutually orthogonal. Then

$$B = \begin{bmatrix} A & A \\ A & -A \end{bmatrix} \tag{1.24}$$

is a 2n x 2n matrix possessing the same properties. This is easily seen by observing that all the rows in $\begin{bmatrix} A & A \end{bmatrix}$ are trivially mutually orthogonal if the rows in A are orthogonal and similarly for the rows of $\begin{bmatrix} A & -A \end{bmatrix}$. Now consider a row, $a_1a_2 \ldots, a_n\ a_1a_2, \ldots, a_n$ from the top half of B, and a second row, $b_1b_2, \ldots, b_n, -b_1 -b_2, \ldots, -b_n$, from the bottom half of B. Then the correlation between the two rows is:

$$\rho = \frac{1}{2n}\left\{ \sum_{i=1}^{n} a_ib_i - \sum_{i=1}^{n} a_ib_i \right\} = 0.$$

Using this technique, it is apparent that, beginning with the code dictionary

$$\begin{matrix} 1 & 1 \\ 1 & -1 \end{matrix}$$

any orthogonal dictionary of 2^n words (n = 1, 2, ...) can be obtained.

Now consider an arbitrary binary orthogonal dictionary of 4t words, each word with 4t symbols. Since the fact that $\sum_{i=1}^{4t} x_i y_i = 0$ implies that $\sum_{i=1}^{4t} x_i(-y_i) = -\sum_{i=1}^{4t} x_i y_i = 0$, it follows that if any word is multiplied by -1, the dictionary is still orthogonal. Thus the dictionary may be normalized by multiplying the appropriate words by -1, so that the initial symbol of each word is +1. If this symbol is now removed, the correlation between any two words is

$$\rho = \frac{1}{N-1} \sum_{i=2}^{N} x_i y_i = \frac{1}{N-1}\left[\sum_{i=1}^{N} x_i y_i - 1 \right] = -\frac{1}{N-1} \qquad (1.25)$$

since $x_1 y_1 = 1$ for any words x and y in the dictionary. Thus, if there exists an orthogonal dictionary of size M = N of N symbol words, then there exists a dictionary of size N of N-1 symbols such that $\rho_{ij} = -\frac{1}{N-1}$ for all $i \neq j$. This is known as a trans-orthogonal code. As was shown earlier, this is the minimum value possible for the maximum correlation between two words of a dictionary of size N. It is readily seen that N words with fewer than N-1 symbols cannot have this property since the correlation, ρ, between two words of length K must be some integral multiple of 1/K. Thus, $k/K = -\frac{1}{N-1}$ implies $k = -\frac{K}{N-1}$, but if $K < N-1$, k cannot be an integer.

It was also shown above that $\rho_{ave} \geq -\frac{1}{N-1}$. This, of course, is achieved by the dictionary described in the previous paragraph since all cross-correlations are equal to $-\frac{1}{N-1}$. In addition, the orthogonal dictionary can be extended to obtain a code of N = 2M M-symbol words achieving this minimum average correlation in a trivial manner, namely by also including in the dictionary the complements of all code words (i.e., those words obtained by replacing all the symbols of each code word by their complements). It was seen that any code word x which is orthogonal to the code word y is also orthogonal to the complement of y. Thus, each word of this doubled orthogonal dictionary, the so-called <u>bi-orthogonal</u> dictionary, has zero correlation with all other words but one -- its complement -- with which its correlation is, of course, -1. Hence,

$$\rho_{\text{ave}} = \frac{(2M-2)\ 0 - 1}{2M - 1} = -\frac{1}{N-1} \ . \qquad (1.26)$$

F. Bi-Orthogonal and Trans-Orthogonal Word Error Probabilities

The error probabilities are also readily determined for trans-orthogonal and bi-orthogonal codes. The trans-orthogonal code word error probability is, from equation 1.20, exactly that of orthogonal code when the signal-to-noise ratio of the latter is increased by a factor of $1 - \rho = \frac{N}{N-1}$. The error probability of the bi-orthogonal code is evidently just:

$$P_c = \int_0^{\infty} p(x)\ P(|y_1|<|x|, |y_2|<|x|, \ldots, |y_{N-1}|<|x|)\ dy_1 dy_2 \ldots dy_{N-1}\ dx \qquad (1.27)$$

$$= \frac{1}{(2\pi)^{N/2}} \int_{-\frac{AE_o^{\frac{1}{2}}}{\sigma_n}}^{\infty} e^{-\frac{s^2}{2}} \left[\int_{-s+\frac{AE_o^{\frac{1}{2}}}{\sigma_n}}^{s+\frac{AE_o^{\frac{1}{2}}}{\sigma_n}} e^{-\frac{t^2}{2}}\ dt \right]^{N-1} ds.$$

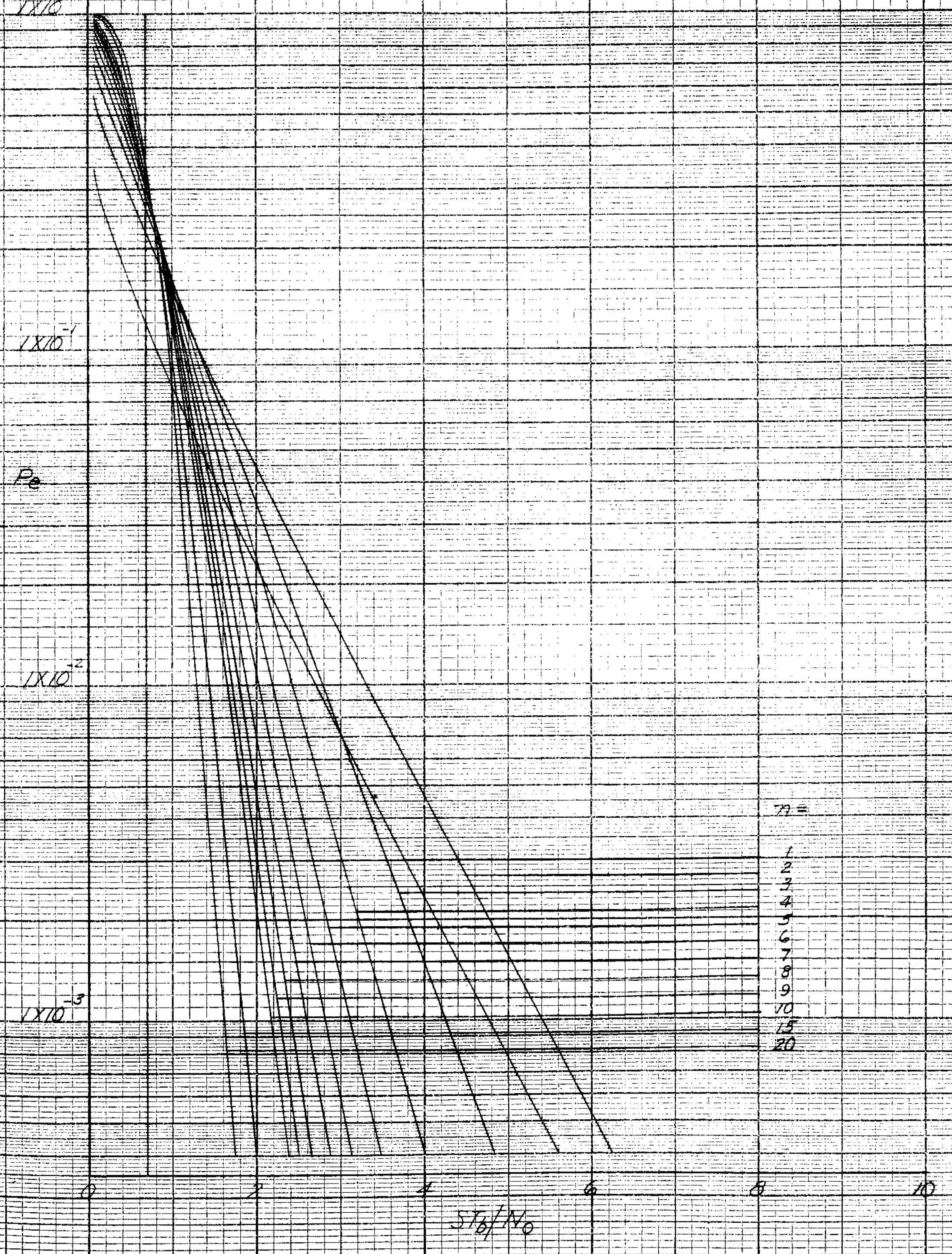
This is apparent upon investigation of equations 1.6 and 1.7 and observing that in this case the error probability is just the probability that the correct correlator output is greater in absolute value than all other outputs and that it correctly represents the sign of the transmitted code word. This value of P_c has also been determined numerically for various values of N and $\frac{ST_b}{N_o}$.(23) The results are plotted in Figure 1.2.

In the pages that follow, the codes that will be considered from a self-synchronizing point of view will be primarily the binary orthogonal codes although the results are easily extendable to the bi-orthogonal situation. The effective $\frac{N}{N-1}$ energy increase inherent in the use of the trans-orthogonal codes is certainly negligible for even moderate values of N, and it is evident that this negative correlation between words is actually a disadvantage if carrier coherence is not available.(24) Excluding additional channels, sign information is not available until synchronization has been obtained (see below). Because of this, it will be seen that negative and positive correlation are equally disadvantageous in the synchronization process. Bi-orthogonal codes, too, suffer from this disadvantage even after synchronization has been obtained because, due to the complete symmetry of the code, it is impossible to determine the sign without some additional information.

G. Bandwidth Occupancy for Binary Orthogonal Codes

It was estimated earlier that the bandwidth occupancy for an orthogonal code consisting of $N = 2^n$ words is $\frac{2^{n-1}}{nT_b}$ where $nT_b = T$ is the time

FIGURE 1.2
BIORTHOGONAL CODES:
WORD ERROR PROBABILITY



necessary to transmit one word. It will now be shown that this is indeed a measure of the bandwidth occupancy of binary orthogonal codes.

Consider the autocorrelation of an infinite sequence of these binary orthogonal waveforms:

$$C(\tau) = \frac{\int_{-\infty}^{\infty} y(t)\, y(t+\tau)\, dt}{\int_{-\infty}^{\infty} y^2(t)\, dt} . \tag{1.28}$$

It is apparent that $C(0) = 1$ and that $C(T) = \frac{1}{N}$ since, in an infinite random sequence of N words, a given word will be correlated with itself $\frac{1}{N}$th of the time. Thus, the autocorrelation function consists of the sum of the following:

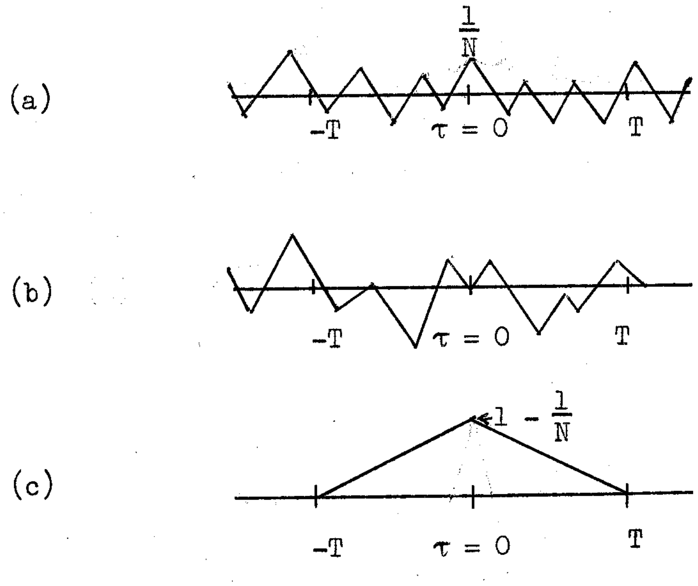


Figure 1.3

where the fine structure of (a) and (b) can be determined for any particular orthogonal code. The component of the autocorrelation function (a) is periodic due to the periodic structure of $C(\tau)$ for $|\tau| > T$; (b) is necessary since, for $|\tau| < T$ the autocorrelation may be different due to the fact that each word is restricted to be compared partially with a phase shift of itself. Finally, (c) results from the contribution at $\tau = 0$ which is necessarily larger than that for any other value of τ.

Regardless of the details of (a) and (b), the power spectral density, obtained by taking the fourier transform of $C(\tau)$, includes a $\left(\frac{\sin \frac{\omega T}{2}}{\frac{\omega T}{2}} \right)^2$ term due to (c) and consequently extends over all finite frequencies. This phenomenon is, of course, identical for all orthogonal codes, binary or not. Although little signal energy is lost if all values of $|\omega| > \frac{2k\pi}{T}$ are suppressed, for $k \geq 3$ for example, resulting in a bandwidth of $\frac{k\ 2^n}{nT_b}$, a more accurate determination of the bandwidth occupancy of these codes is afforded by the following considerations: In the proposed system, the subcarrier, $\sin \omega_o t$, where $\omega_o = \frac{m\pi}{2T_S} = \frac{mN\pi}{2T}$ for some integer m, is to be phase modulated by 0^o or 180^o corresponding to the occurrence of a 0 or a 1, respectively, in the binary code word. Now suppose there is a second system phase coherent with the first but operating at a frequency $\sin(\omega_o + 2\pi f)\ t$. The correlator for the first system, upon receiving a signal from the second, forms the integration:

$$\int_{o}^{T/N} \sin(\omega_o t + \phi_1) \sin((\omega_o + 2\pi f) t + \phi_2) dt$$

$$= \int_{o}^{T/N} \cos(2\pi f t + (\phi_1 - \phi_2))dt - \int_{o}^{T/N} \cos((2\omega_o + 2\pi f) t + \qquad (1.29)$$

$$(\phi_1 + \phi_2)) dt = \frac{1}{2\pi f} \sin(\frac{2\pi f T}{N} + \phi_1 + \phi_2) -$$

$$\frac{1}{2\omega_o + 2\pi f} \sin((m\pi + \frac{2\pi f T}{N}) + (\phi_1 + \phi_2)).$$

But ϕ_1, $\phi_2 = 0$ or π and hence if $f = \frac{kN}{2T}$ for any non-zero integer k, this correlation is identically zero. Consequently, identical communication systems can be operated independently, without mutual interference, at all frequencies separated by some multiple of $\frac{N}{2T}$. The effective bandwidth occupancy is thus $\frac{N}{2T} = \frac{2^{n-1}}{nT_b}$ as predicted.

H. Some Comments on Mechanization

As was mentioned earlier, one of the advantages of digital coding systems is in the ease of their mechanization. If the code is a binary group code (cf. Chapter 3), each code word is formed as one of the 2^n possible linear combinations (over GF(2)) of n generators with the possible addition of an n + 1st element, the coset leader, to each code word. The data word may be used as one of the inputs to an "and" gate causing the corresponding element to be added or not added to form the complete code word. Such a system is shown in Figure 1.4 where $d_1, d_2, \ldots, d_n$ represent the data bits and $x_1^i, x_2^i, \ldots, x_N^i$ the symbols of the ith generator. The resultant code word is then used to phase modulate a subcarrier as described above.

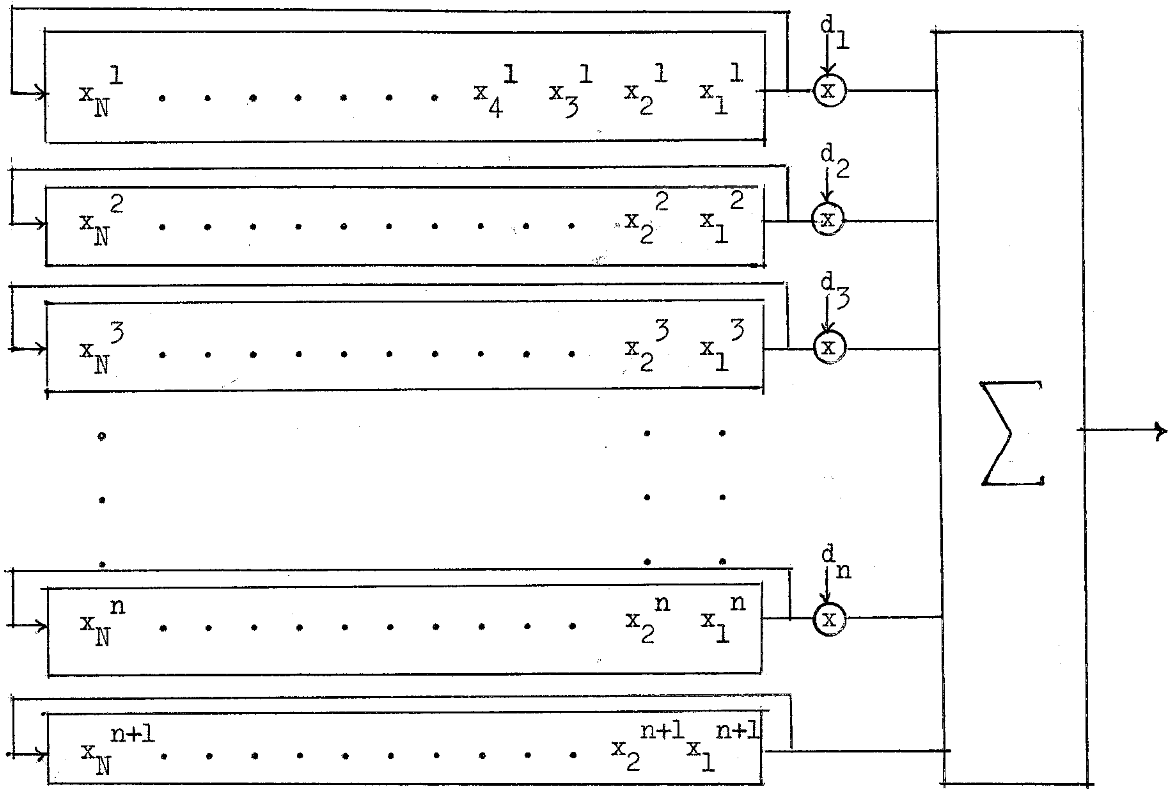

$x_N^1 \ldots x_4^1\ x_3^1\ x_2^1\ x_1^1$
$x_N^2 \ldots x_2^2\ x_1^2$
$x_N^3 \ldots x_2^3\ x_1^3$
$x_N^n \ldots x_2^n\ x_1^n$
$x_N^{n+1} \ldots x_2^{n+1} x_1^{n+1}$
d_1
d_2
d_3
d_n
x
$\sum$

Figure 1.4

The decoder is also vastly simplified by the use of digital codes. A matched filter is generally difficult to construct. The correlator mechanization of the matched filter, while readily realizable, becomes unwieldy for moderately large values of N since a correlator is, in general, required for each code word. The digital code, however, needs only one correlator which integrates over each symbol time. The output is sampled at the end of this time and converted to digital form. The N such results, corresponding to the duration of one word, are added and subtracted in accordance with whether the local word symbols, y_i^j , are ones or zeros, respectively. The decision device then selects the largest of these sums to determine the received code word. All processing after the correlator is done digitally either by a general purpose or a special purpose digital computor. Such a decoder is shown schematically in Figure 1.5:

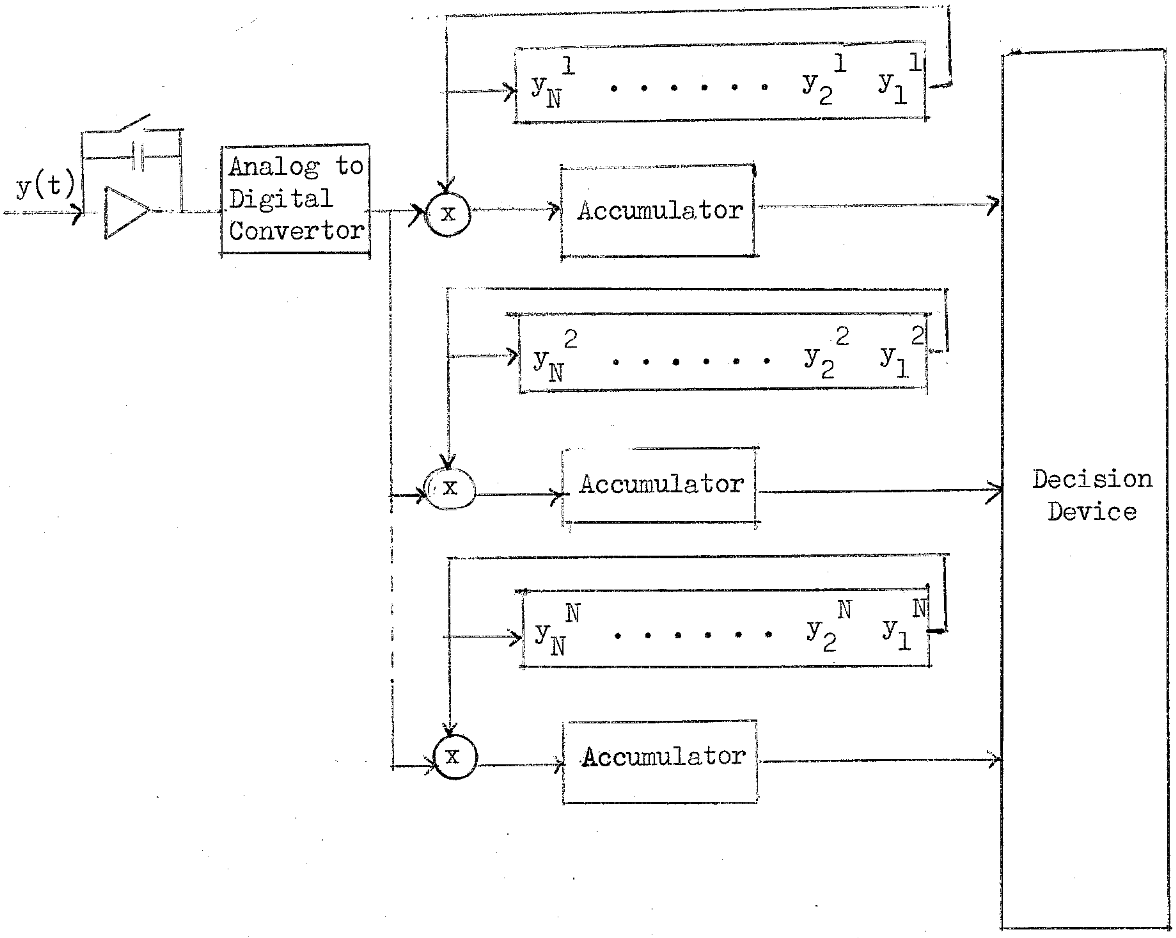

y(t)
Analog to Digital Convertor
y_N^1 . . . . y_2^1 y_1^1
x
Accumulator
y_N^2 . . . . y_2^2 y_1^2
x
Accumulator
y_N^N . . . . y_2^N y_1^N
x
Accumulator
Decision Device

Figure 1.5

I. Symbol Synchronization

As mentioned earlier, the number of independent positions which need to be investigated in order to determine the correct word synchronization is indicated by the value $2WT = N$. It is an additional advantage of binary codes that this quantization of the number of positions to be investigated is automatically assured once the knowledge of the instants of time when the symbol can change value is known. It shall be assumed, in fact, that this symbol synchronization has been obtained before word synchronization is attempted. For example, if the subcarrier consists of a sine wave of period $2T/N$ modulated by 0° or 180° at every zero crossing corresponding to whether the code symbol is one or zero, the symbol synchronization information is available as soon as the phase of the subcarrier has been determined. The latter may be done by squaring the subcarrier and detecting the double frequency component with a phase-locked loop. Note that there is a 180° phase ambiguity here. It is apparent that this ambiguity will always be present in a one channel space telemetry system so long as both binary code symbols are represented by equal absolute amplitudes and time intervals. For this reason it is impossible to determine which symbol corresponds to a one and which corresponds to a zero. The existence of the complement of every code word in the bi-orthogonal dictionary prohibits this distinction even after word synchronization. Hence, as was mentioned above, these codes cannot be used unless this ambiguity in sign can somehow be resolved by other means.

Chapter 2

SYNCHRONIZATION USING THE PROPERTIES OF A RANDOM SOURCE

A. The Probability of Correct Synchronization

Any statistical phenomenon whose expected value differs in synchronous operation from that in non-synchronous operation may be used to distinguish between the two cases if the noise has finite variance and enough observations are made. Such a phenomenon will be investigated here under the assumption that the source is completely random.

The procedure to be discussed is equally applicable to any coset of any orthogonal group code. It is shown in Chapter 3 that one column of the matrix of +1's and -1's representing an orthogonal group coset contains all positive (or all negative) elements, while all other columns contain $\frac{1}{2}$ positive and $\frac{1}{2}$ negative elements. Identifying the occurrence of an element from this unique column obviously establishes word synchronization. The method will consist of forming the integral:

$$I_{k/N} = \int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T} \left[x(t) + n(t) \right] dt \qquad (2.1)$$

where x(t) refers to the received binary sequence and n(t) to the additive white Gaussian noise. The "phase" k, where k can be any integer from zero to N-1, represents a possible starting symbol for the code word. Without loss of generality, the unique column mentioned above may be identified with the phase k = 0. Now evaluate the integral $I_{k/N}$ for each value of k of the received sequence and observe that:

$$E(I_{k/N}) = \int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T} E\left[x(t)\right] + E\left[n(t)\right] dt = \begin{cases} 0 & k \neq 0 \\ \frac{AT}{N}^{*} & k = 0 \end{cases} \qquad (2.2)$$

since $E\left[n(t)\right] = 0$ by assumption, and:

$$E\left[x(t)\right] = P_r(x = A) \cdot A + P_r(x = -A) \cdot (-A) = \begin{cases} 0 & k \neq 0 \\ A & k = 0 \end{cases}$$

Further,

$$E(I^2_{k/N}) = E \int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T} \int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T} \left[x(t) + n(t)\right]\left[x(u) + n(u)\right] dtdu$$

[*] Note that the relevant parameter here is the <u>received</u> signal energy

$$E_r = \int_0^T A^2x^2(t)\,dt = A^2E_o = A^2B^2T = ST$$

where ±B is the amplitude of the transmitted signal and A the factor by which this amplitude is reduced during transmission. Since B, in itself, is unimportant, it will be convenient to let B = 1 in this and succeeding chapters and let A represent the amplitude of the received signal rather than the relative amplitude.

$$= \mathbf{E}\left[\int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T} x(t)\ \ dt\right]^2 + 2\int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T}\int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T} E\left[x(t)\right]\ E\left[n(u)\right]\ dtdu$$

$$+ \int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T}\int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T} E\left[n(t)\ n(u)\right]\ dtdu$$

$$= \int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T}\int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T} A^2\ dtdu + \int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T}\int_{\frac{k\,T}{N}}^{\frac{k+1}{N}T} \sigma_n{}^2\,\delta(t-u)\ dtdu$$

$$= \frac{A^2T^2}{N^2} + \sigma_n{}^2\ \frac{T}{N}\ . \qquad (2.3)$$

Thus

$$\begin{aligned}
\mu_{k=0} &= \frac{AT}{N} \\
\mu_{k\neq 0} &= 0 \\
\sigma^2_{k=0} &= E(I^2) - \left[E(I)\right]^2 = \sigma_n^2\ \frac{T}{N} \\
\sigma^2_{k\neq 0} &= \frac{A^2T^2}{N^2} + \sigma_n^2\ \frac{T}{N}\ .
\end{aligned} \qquad (2.4)$$

Since the noise is assumed to be white and therefore uncorrelated, and since the code words to be transmitted presumably occur randomly, the value of I at (i + k/N)T and that at (j + k/N)T are statistically independent for $i \neq j$. Applying the Central Limit Theorem to the sum $s_k = \sum_{i=1}^{m} I_{i+k/N}$, one obtains a random variable s_k, the density function of which is asymptotically normal with a standard deviation $\sigma'_k = m^{\frac{1}{2}} \sigma_k$ and mean $\mu'_k = m\mu_k$.

The probability of correctly identifying the value k = 0 and, in addition, correctly resolving the sign ambiguity mentioned above, is then the probability that the particular value of s_o so obtained is positive (or negative) and greater in absolute magnitude than that of all other s_k, $k \neq 0$. Thus,

$$P_s = \int_{o}^{\infty}\int_{-s_o}^{s_o} \cdots \int_{-s_o}^{s_o} p(s_o, s_1, \ldots, s_{N-1}) ds_1 \ldots ds_{N-1}\, ds_o \tag{2.5}$$

where $p(s_o, s_1, \ldots s_{N-1})$ represents the joint probability density of the random variables, s_i, (i = 0, 1, ... N-1). Although the noise contributions to s_i are statistically independent of those contributions to s_j, for $i \neq j$, the signal contributions are not independent, since they consist of different symbols from the same words. Observe, however, that for any value of α,

$$P_s \geq \int_{\alpha}^{\infty}\int_{-\alpha}^{\alpha} \cdots \int_{-\alpha}^{\alpha} p(s_o, s_1, \ldots, s_{N-1})\, ds_1\, ds_2 \ldots ds_{N-1}\, ds_o$$

$$\begin{aligned} &= P(s_o > \alpha, |s_1| < \alpha, |s_2| < \alpha, \ldots, |s_{N-1}| < \alpha) \\ &= 1 - P(s_o < \alpha \cup |s_1| > \alpha \cup \ldots \cup |s_{N-1}| > \alpha) \qquad (2.6) \\ &\geq 1 - \left[P(s_o < \alpha) + P(|s_1| > \alpha) + \ldots + P(|s_{N-1}| > \alpha) \right]. \end{aligned}$$

The last step follows from the well-known inequality

$$P(A_1 \cup A_2 \ldots \cup A_n) \leq P(A_1) + P(A_2) + \ldots + P(A_n).$$

Now

$$\begin{aligned} P(s_o < \alpha) &= \frac{1}{(2\pi)^{\frac{1}{2}} m^{\frac{1}{2}} \sigma_o} \int_{-\infty}^{\alpha} \exp\left\{ - \frac{(s_o - m\mu_o)^2}{2m\,\sigma_o^{\;2}} \right\} ds_o \qquad (2.7) \\ &= \frac{1}{(2\pi)^{\frac{1}{2}}} \int_{-\infty}^{-(1-\lambda)m^{\frac{1}{2}}R^{\frac{1}{2}}\left(\frac{2\log_2 N}{N}\right)^{\frac{1}{2}}} e^{-\frac{t^2}{2}}\, dt \end{aligned}$$

where $\alpha \equiv \lambda\, m \frac{AT}{N}$ and $R = \frac{ST_b}{N_o} = \frac{A^2 T}{2\,\sigma_n^{\;2}\,\log_2 N}$ the ratio of the signal energy per bit of information to the noise power per unit bandwidth, defined in Chapter 1. Similarly,

$$\begin{aligned} P(|s_i| > \alpha) &= 1 - \frac{1}{(2\pi)^{\frac{1}{2}}} \int_{\frac{-\alpha - m\mu_i}{m^{\frac{1}{2}}\sigma_i}}^{\frac{\alpha - m\mu_i}{m^{\frac{1}{2}}\sigma_i}} e^{-\frac{t^2}{2}}\, dt \\ &= 1 - \frac{1}{(2\pi)^{\frac{1}{2}}} \int_{-\beta}^{\beta} e^{-t^2/2}\, dt \end{aligned}$$

where $\beta = \lambda \left[\dfrac{m\, R(\frac{2\log_2 N}{N})}{R(\frac{2\log_2 N}{N}) + 1} \right]^{\frac{1}{2}}$.

Selecting a suitable value of λ, one obtains the following table giving an upper bound on the number of words, m, necessary to establish synchronization with a probability, P_s, of .999 and .9999 for various sizes of orthogonal codes.

Table 2.1

Upper Bounds on
Synchronization Times Assuming a Random Source

$N = 2^n$	R	λ	m(.999)	mT(secs.)	m'(.9999)	m'T(secs.)
8	4.00	.67	45	3	57	3.8
16	3.25	.65	62	5.5	80	7.1
32	2.88	.64	95	10.6	124	13.8
64	2.50	.62	165	22	220	29.3
128	2.40	.614	280	43.6	380	59.1

The value of R in the above table has been chosen to correspond to a probability of a word error of approximately 10^{-3} in normal synchronous operation and the word time T has been selected so that information is transmitted at the standard teletype rate of 45 bits/second.

B. Asymptotic Results

The convergence time is certainly not prohibitive for the smaller dictionaries. However, the time necessary grows rather rapidly with N. It is readily apparent, in fact, that m must grow asymptotically more rapidly than $\frac{N}{\log_2 N}$. This may be seen by an investigation of equations 2.6, 2.7 and 2.8. That is,

$$P_s \geq 1 - P(s_o < \alpha) - (N-1)\, P(|s_1| > \alpha) \qquad (2.6a)$$

and it is thus necessary that the term, $P(s_o < \alpha) + (N-1)\, P(|s_1| > \alpha)$, does not increase with N. But for moderate values of N, the denominator in the term β of equation 2.8 is effectively 1, and hence the numerators in the limits of the integration of both equation 2.7 and equation 2.8 must not decrease as N increases if the values of $P(s_o < \alpha)$ and $P(|s_1| > \alpha)$ are not to increase. Since, for a fixed error probability, R is a decreasing function of N, the statement concerning the rate of increase of m as a function of N follows.

It should be remembered, too, that these time estimates depend upon the assumption of a random distribution of the incoming sequence. If, for example, the source is such that only one-half the dictionary words are sent during the transmission, then, as investigation of the codes under discussion reveals (see Chapter 3), there is another value of k, other than k = 0, for which every symbol is either always +A or always -A. Considering these two positions alone lowers the probability of correct synchronization to $\frac{1}{2}$, regardless of the number of words observed. In general, if the source limits its selection to any subgroup of order 2^{n-p} of the dictionary group, then there are 2^p-1 values of $k \neq 0$ for which the above is true. This becomes more serious for large dictionaries for two reasons: (1) Since there are more words from which to choose, the possibility that a sizeable subset of the words will not be sent by a non-random selection increases, and (2) the number of different words sent in any given time decreases if the bit rate is held constant. The next section will be concerned with a possible method of overcoming these difficulties inherent in this type of synchronization.

C. An Alternate Synchronization Technique

Before proceeding, however, another method should be mentioned which may be shown to be equivalent to the synchronization scheme just discussed, when applied to the same dictionaries, so far as the probability $P_s(m)$ is concerned, but which has a somewhat different mechanization. Here the quantity to be investigated, corresponding to

$$\int_{(i + k/N)T}^{(i + \frac{k+1}{N})T} \left[y(t) + n(t)\right] dt$$

in the previous case, is:

$$Z_i(k) = \frac{1}{N} \sum_{\ell = 1}^{N} Z_{i\ell}(k) = \frac{1}{N} \sum_{\ell = 1}^{N} \int_{(i + k/N)T}^{(i + 1 + k/N)T} \left[y(t) + n(t)\right] x_\ell(t)\, dt \tag{2.9}$$

where $x_\ell(t)$ represents the ℓ^{th} locally generated dictionary word. Thus all correlations are performed for each k and summed. The statistics, as already mentioned, are the same for $\sum_{i=1}^{m} Z_i(k)$ as they were for the s_k discussed above when group codes or cosets of group codes are used. Non-group orthogonal codes may actually be found which decrease the value corresponding to σ_k above for some k. However, the improvement seems to be slight, and the analysis considerably more difficult for these codes. The main reason for mentioning this process is that it is similar to that used in Chapter 5 and may, in fact, be used concurrently with it with only slightly more computing operations. Its obvious disadvantage is that more computation is involved here than in the previous method.

Chapter 3

COMMA-FREE CODES

A. Comma Freedom and Its Importance

In normal operation, the correlations

$$C_{io} = \int_{o}^{T} x_i(t)\left[x_r(t) + n(t)\right] dt \quad (i = 1, 2, \ldots, N) \qquad (3.1)$$

are evaluated, and the largest of these integrals determines which of the N signals was most likely sent. If synchronization has not been obtained, there are N(N-1) other correlations of the form

$$C_{ik} = \int_{o}^{T} x_i(t)\left[y_k(t) + n(t)\right] dt \quad \begin{matrix}(i = 1, 2, \ldots, N) \\ (k = 1, 2, \ldots, N-1)\end{matrix} \qquad (3.2)$$

where $y_k(t)$ is a sequence formed by the last N-k symbols of one code word followed by the first k symbols of a second code word. If there exists a code for which the value of C_{io} and C_{ik} are significantly different, regardless of the sequence transmitted, and if this difference is suitably exploited, a new method for synchronization may be available.

For a particular sequence $y_k(t)$, and for stationary white Gaussian noise, n(t), the random variable C_{ik} is Gaussian with an expected value

$$E(C_{ik}) = \int_{o}^{T} x_i(t)\; y_k(t)\; dt = \frac{AT}{N} \sum_{j} x_j^i\; y_j^k \qquad (3.3a)$$

and variance

$$E(C_{ik}^2) - \left[E(C_{ik})\right]^2 = \int_0^T \sigma_n^{\,2}\, x_i^{\,2}(t) = \sigma_n^{\,2} T \tag{3.3b}$$

where x_j^i and y_j^k , of course, represent the j<u>th</u> symbols of the sequences of +1's and -1's which generate the waveforms $x_i(t)$ and $y_k(t)$. Note that when $k = 0$, $y_o(t)$ is just $x_r(t)$ for some r.

If the vectors x^i are normalized by dividing each component by $\sqrt{N}$ then, since $x^i \cdot x^j = \delta_{ij}$, they may be considered as an orthonormal basis in N-dimensional space over the complex field. Any other vector y^k can then be written as a linear combination of the vectors x^i:

$$y_j^k = \sum_{i=1}^{N} a_i\, x_j^i \tag{3.4}$$

where the subscript j denotes the j<u>th</u> component of the vector. The sequence represented by y^k is then completely characterized by the vector $a(k) = \left\{ a_i \right\}$ whose components can be obtained by correlating y^k with the N code vectors x^i. That is:

$$\frac{1}{A\,\frac{T}{N}}\, E(C_{ik}) = x^i \cdot y = \sum_{j=1}^{N} x_j^i\, y_j = \sum_{j=1}^{N} x_j^i \sum_{\mu=1}^{N} a_\mu\, x_j^\mu$$

$$= \sum_{\mu=1}^{N} a_\mu \sum_{j=1}^{N} x_j^i\, x_j^\mu = \sum a_\mu\, \delta_{i\mu} = a_i. \tag{3.5}$$

Obviously, the vector $a(k = 0)$ is just one of the unit vectors e^i, all of whose components are zero except the i<u>th</u> which has the value one.

A further condition on the vector, a(k), can be seen by noting that if y is to represent a sequence of $\pm \frac{1}{\sqrt{N}}$'s, then

$$\sum_{j=1}^{N} y_j^{\,2} = 1 = \sum_{j=1}^{N} \sum_{\mu=1}^{N} a_\mu x_j^\mu \sum_{\upsilon=1}^{N} a_\upsilon x_j^\upsilon$$

$$= \sum_{\mu=1}^{N} \sum_{\upsilon=1}^{N} a_\mu a_\upsilon \delta_{\mu\upsilon} = \sum_{\mu=1}^{N} a_\mu^2 = 1. \tag{3.6}$$

It would appear to be desirable, in view of equation 3.3, to impose a further restriction, if possible, on the code dictionary which would somehow maximize the difference between a(k ≠ 0) and a(0). To do this, the term "difference" must be defined. The optimum definition, in turn, depends upon the method in which this difference is to be used to facilitate synchronization. An intuitively reasonable measure of the difference between a(k) and any of the in-phase vectors $\pm e^i$ is the "mean square error":

$$d = \min_i \sum_{j=1}^{N} (a_j \pm e_j^i)^2$$

$$= \min_i \sum_{j=1}^{N} a_j^{\,2} - 2|a_i| + 1 \tag{3.7}$$

$$= \min_i 2(1 - |a_i|) = 2(1 - \max_i |a_i|).$$

Note that the choice in sign results from the inherent sign ambiguity mentioned above and must be made so that a(k) is compared to the "closest" of the unit vectors $\pm e^i$. Thus, maximizing the difference d according to this definition is equivalent to minimizing the component of a with the maximum absolute value. This criterion is, in fact, consistent with the synchronization scheme which will be described in Chapter 5.

It will be recalled from Chapter 2 that the correlation between two vectors, x and y, may be expressed as

$$C_{xy} = \frac{A - D}{N} \tag{3.8}$$

where A is the number of agreements and D is the number of disagreements between corresponding components of the two vectors. It is evident that for a given N, the correlation is uniquely determined if either A or D is specified. Further, observe that if the components of the vectors x and y are represented by 0's and 1's, then D is just the number of ones in their modulo two term-by-term sum. That is, in terms of the so-called "Hamming distance" (the number of ones in a vector, z, of zeros and ones, written $|z|$), D is simply equal to $|x + y|$, where the + denotes the sums of the corresponding components over the binary field. Several obvious identities will be useful:

$$
\begin{aligned}
|x + y| &= |y + x| \\
|x + y| &= |x + I + y + I| = |\overline{x} + \overline{y}| \\
|x + x| &= 0 \\
|x + \overline{y}| &= |x + y + I| = |\overline{x} + y| = N - |x + y| \\
|x + \overline{x}| &= |x + x + I| = |I| = N
\end{aligned}
\tag{3.9}
$$

where I is the vector consisting of all ones.

It is convenient at this point to refer to the concept of comma freedom. If a dictionary of N-symbol words is comma-free, then by definition, any series of 2N-1 symbols occurring in an arbitrary sequence of these words must contain a unique dictionary word of length N. That is, the correlation between a dictionary word and any word formed from the combination of two words must be such that the number of disagreements D be greater than zero. A useful generalization of this condition, applicable to the situation here, may be obtained by the following considerations: d in equation 3.7 is maximized when $\max_i |a_i| = \max_i |C_{yx^i}| = \max_i \left|1 - \frac{2D_i}{N}\right|$ is minimized. Since specifying that $p \leq D_i \leq N-p$ is equivalent to requiring that $\left|\frac{2p}{N} - 1\right| \leq \max_i |a_i| \leq \left|1 - \frac{2p}{N}\right|$, it is seen that $\max_i |a_i|$ is minimized by maximizing the value of $p \leq \frac{N}{2}$ in the above expression. We shall refer to a code in which D_i satisfies these inequalities for any sequence of N symbols not actually forming a word as a comma-free code of <u>index</u> p. An optimum orthogonal code of size N, then, will be one in which the index of comma freedom is maximized.

B. The Number of Distinct Orthogonal Dictionaries of N Words

Before proceeding with the discussion of optimum codes, it would be useful to determine the number of different orthogonal codes which may be considered. If, for example, the number of different dictionaries is moderately small, perhaps they can all be systematically listed and investigated for comma freedom and the optimum dictionary selected. Two dictionaries are considered different if at least one word in the first is different from all words in the second. It is observed that if a dictionary is written as a square matrix of ones and zeros, the words of the dictionary determining the rows, then no permutation of the rows forms a different dictionary. If one of the rows is complemented, however, a new dictionary may result. It is still an orthogonal dictionary since any word orthogonal to another word is also orthogonal to its complement. That is, if

$$|x + y| = \frac{N}{2}$$

then

$$|x + y + I| = N - \frac{N}{2} = \frac{N}{2}.$$

In addition, since the number of disagreements between two binary sequences remains the same if both sequences undergo identical permutations, or if the ith symbol in both sequences is complemented, the dictionary obtained by permutation or complementation of any of the columns of any orthogonal dictionary is still orthogonal, although the two may be different. Any two dictionaries, one of which may be obtained by some complementation or permutation of the columns and rows of the other, are said to be in the same equivalence class. A lower bound on the number of orthogonal dictionaries may be obtained by evaluating the size of one equivalence class. This, in fact, will be done below. The class to be investigated is that containing those code dictionaries whose words form groups under the operation of symbol-by-symbol modulo-two addition. That is, the following axioms are satisfied by the dictionary words under this operation:

(1) The group (dictionary) contains an identity element ϵ such that $x + \epsilon = x$. ϵ is obviously the vector containing all zeros.

(2) $x_1 + x_2 = x_3$ where x_3 is in the group if x_1 and x_2 are in the group.

(3) The group includes an inverse $-x$ for any x in the group such that $x + (-x) = \epsilon$. In this case, $-x$ is seen to be the element x itself.

(4) $x + (y + z) = (x + y) + z$ for any x, y and z in the group. Further, since $x + y = y + x$, the group is said to be Abelian. An example of such a group is the following:

$\epsilon = 0\;0\;0\;0$

$x_1 = 0\;1\;0\;1$

$x_2 = 1\;1\;0\;0$

$x_3 = 1\;0\;0\;1.$

Note that $x_1 + x_2 = x_3$ and that $x_i + x_i = \epsilon$ and $x_i + \epsilon = x_i$ for all i.

An elementary theorem in group theory states that the order (number of words or elements) of a sub-group must divide the order of the group. Thus, since the orthogonal group dictionary of size N must be a sub-group of all 2^N possible vectors of N binary symbols, N must be a power of two. Further, it is easily shown, using axiom (2), that there are exactly n <u>generators</u> $x_1, x_2, \ldots, x_n$ such that any word of the group of order $2^n = N$ can be written as $x_j = a_1x_1 + a_2x_2 + \ldots + a_nx_n$ where a_i is either zero or one. These generators may be selected in the following manner: x_1 can be any member of the group but the identity ϵ, x_2 any element except x_1 or ϵ, x_3 any element except ϵ, x_1, x_2 or $x_1 + x_2$, etc. These combinations of the n generators obviously form at most 2^n elements. That these elements are distinct follows immediately from the manner in which the generators were chosen. The following theorem will be useful:

<u>THEOREM</u>: Any binary group code is orthogonal if, and only if, all its elements, except the identity, contain exactly one half ones and one half zeros.

<u>Proof</u>: Labeling the group G, one observes that for all $g_1 \epsilon G$, and $g_2 \epsilon G$, $|g_1 + g_2| = 2^{n-1}$ in order to insure orthogonality, since the Hamming distance between two orthogonal elements of length 2^n must be 2^{n-1} (cf. equation 3.8). But $|g_1 + g_2| = |g_3|$, $g_3 \epsilon G$. Thus g_1 and g_2 are orthogonal if, and only if, g_3 contains one half ones and one half zeros. But the sum of any two distinct elements in G is still in G and is not the identity and, since $g_1 + g_2 = g_3$ implies that $g_1 + g_3 = g_2$ and that $g_2 + g_3 = g_1$, any element in G, except the identity, can be written as the sum of two other distinct elements in G. The identity, of course, is orthogonal to any element containing one half ones and one half zeros.

Thus to count the number of orthogonal group codes, it is sufficient to enumerate the number of ways the n generators of G can be selected so that all elements thus generated contain one half ones and one half zeros. Any element containing half zeros and half ones is acceptable for the first generator. For convenience, a permutation operator may be selected which shifts all the ones to the right half of the element: 00 ..... 011 ..... 1. The second generator must, of course, also contain half ones and half zeros and be such that the element produced by it and the first generator contains half ones and half zeros. Now assume that the first half of this element contains υ ones and the second half $2^{n-1}-\upsilon$ ones. Then the number of ones in the sum of the two elements is $\upsilon + 2^{n-1} - (2^{n-1} - \upsilon) = 2\upsilon$ since the summation simply inverts the last half of the second element. Thus $\upsilon = 2^{n-2}$. Since a permutation which interchanges only those binary digits found in the same half of the elements does not alter the identity or the first element, the second element may be permuted to the following form:

00 ..... 011 ..... 100 ..... 011 1

where each block is of length 2^{n-2}, and the first element is left invariant.

This process can be extended in a straightforward manner. Assume that the first m of the n generators have been obtained, and that they are necessarily of the form such that for each k the following operations may be performed. The k<u>th</u> generator is divided into 2^k "cells". Beginning from the left, the first cell consists of 2^{n-k} zeros, the second 2^{n-k} consecutive digits being ones, etc. Since the k + 1st

permutation operator interchanges digits within these 2^k cells only, and never between them, all previous generators are not altered by these permutations and the generators are equivalent, under permutation, to

```
000000 ..................... 0000
00 .......... 00111 .......... 11
00 .. 00111 ... 1100 . 00111 .. 11
etc. .
```

Now divide each of these generators into 2^m cells corresponding to the 2^m blocks of all zeros or all ones of the mth generator. Each set of 2^{n-m} digits comprising the cells of the first m generators then consists entirely of ones or entirely of zeros. Since the group generated by these generators is orthogonal, by assumption, the group obtained by replacing each of the cells of 2^{n-m} ones by single ones and similarly for the cells of zeros is also orthogonal. This follows trivially from the definition of orthogonality since the number of agreements and the number of disagreements are each reduced by the same factor, $\frac{1}{2^{n-m}}$. The resulting array of ones and zeros has the same form as above except that the last row is now 010101 .... 01. Designate this $2^m \times 2^m$ orthogonal array by A.

In order that a new generator produce elements which are orthogonal to each of the elements in the original group of 2^m elements, the number of ones in each resulting element must be 2^{n-1}. This is clearly equivalent to requiring that

$$A \begin{bmatrix} p_1 \\ \cdot \\ \cdot \\ \cdot \\ p_{2^m} \end{bmatrix} = \begin{bmatrix} 2^{n-1} \\ \cdot \\ \cdot \\ \cdot \\ 2^{n-1} \end{bmatrix}$$

where p_i is the number of ones in the ith cell of the $m + 1^{st}$ generator, $0 \circ p_i = p_i$, $1 \cdot p_i = 2^{n-m} - p_i$, and A is as defined above. Since each row of A except the first consists of half ones and half zeros, there are 2^{m-1} occurrences of the term 2^{n-m} on the left side of each of the above linear equations except the first. Bringing these terms to the right side, and designating by A' the matrix resulting when the ones of A are replaced by -1's and the 0's by ones, one obtains:

$$A'p = \begin{bmatrix} 2^{n-1} \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix}$$

where ordinary multiplication is now intended. Since $B = \frac{A'}{2^{m/2}}$ is a unitary matrix:

$$p = \frac{1}{2^{m/2}} B^T \begin{bmatrix} 2^{n-1} \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix} = \frac{2^{n-1}}{2^{m/2}} \begin{bmatrix} \frac{1}{2^{m/2}} \\ \frac{1}{2^{m/2}} \\ \cdot \\ \cdot \\ \frac{1}{2^{m/2}} \end{bmatrix} = \begin{bmatrix} 2^{n-m-1} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ 2^{n-m-1} \end{bmatrix}$$

Thus each cell contains 2^{n-m-1} ones and 2^{n-m-1} zeros. It now follows by induction that all the generators may be selected and successively permuted into the following array:

00................00................00

000...............01................11

000......01.......110.......001......11

00..01...10..01...10...01...10...01....1

etc., where the pth generator contains alternating blocks of 2^{n-p} zeros and 2^{n-p} ones. Any set of generators may be obtained by the inverse of the product of the permutations necessary to arrange them in the above fashion. Thus, all orthogonal groups are in the same equivalence class.

The number of ways these generators may be selected is as follows: The first generator may be chosen from any element of 2^n digits, exactly 2^{n-1} of which are ones; i.e., in $\binom{2^n}{2^{n-1}}$ ways. The second may be chosen from any element with 2^n digits, the half corresponding to the ones of the first element containing 2^{n-2} ones and 2^{n-2} zeros and similarly for the other half. Thus, there are $\binom{2^{n-1}}{2^{n-2}}^2$ such elements. Continuing this argument, the pth element may be chosen from any of the $\binom{2^{n-p+1}}{2^{n-p}}^{2^{(p-1)}}$ possible elements satisfying the necessary constraints. This is continued until all n generators are selected.

Now, however, a given group can be generated from more than one set of generators. In particular, given a group, the first generator can be selected from any of the 2^n-1 elements where the identity is, of course, excluded. The second can be selected from any of the 2^n-2 elements from which the first generator and the identity are excluded.

The third may be selected from any of the 2^n-2^2 elements not generated by the first two generators, etc. The number of ways in which the generators may be selected from the total group of elements of 2^n binary digits must then be divided by the number of ways in which the generators of a particular sub-group may be selected; that is by

$$(2^n - 2^0)\ (2^n - 2^1)\ (2^n - 2^2) \dots (2^n - 2^{n-1})$$

Thus the following theorem has been proved.

THEOREM: The total number of distinct orthogonal binary groups is:

$$\frac{\binom{2^n}{2^{n-1}} \binom{2^{n-1}}{2^{n-2}}^2 \binom{2^{n-2}}{2^{n-3}}^4 \binom{2^{n-3}}{2^{n-4}}^8 \dots \binom{2^1}{2^0}^{2^{n-1}}}{(2^n - 2^0)\ (2^n - 2^1) \dots (2^n - 2^{n-1})} \qquad (3.10)$$

$$= \frac{2^n!}{(2^n - 2^0)\ (2^n - 2^1) \dots (2^n - 2^{n-1})} \ .$$

Corollary: The number of distinct bi-orthogonal groups is:

$$B(2^n) = \frac{(2^n)!}{2^n(2^n - 2^0)\ (2^n - 2^1) \dots (2^n - 2^{n-1})} \ . \qquad (3.11)$$

Proof: A bi-orthogonal group is generated by the same elements as an orthogonal group with the addition of the element I containing ones in every position. Given the n+1 generators of such a group, the n generators of an orthogonal group may be selected from them in 2^n ways since any of the n generators, x_i, excluding I, or its complement, $x_i + I$, may be selected. No orthogonal group can include both x_i and $x_i + I$ and hence all 2^n orthogonal groups selected from one bi-orthogonal group are distinct. Two orthogonal groups selected from different bi-orthogonal groups cannot possibly have identical generators and hence are also necessarily distinct. Thus there are 2^n orthogonal groups for every bi-orthogonal group. This completes the proof.

Finally, one can prove:

THEOREM: There are at least

$$S(2^n) = \frac{(2n)!\,(2^{2^{n+1}}-2n-1)}{(2^n - 2^0)\,(2^n - 2^1)\,(2^n - 2^2)\,\ldots\,(2^n - 2^{n-1})} \qquad (3.12)$$

distinct binary orthogonal sets of length $L = 2^n$. To prove this recall an elementary theorem of group theory which states that, for each subgroup of order 2^{n+1} of the group of order 2^{2^n} (formed by all binary vectors of 2^n components), there are $2^{2^n-(n+1)}$ distinct cosets. (A group coset is obtained here by adding, modulo two, a fixed binary N-tuple to every element of the group. This is, of course, equivalent to complementing those columns corresponding to the components of this N-tuple which are ones.) Further, no two distinct groups can have any cosets in common. Now consider the number of ways an orthogonal set can be selected from a bi-orthogonal coset, noting that any orthogonal set in the group equivalence class must form a bi-orthogonal coset when the complements of all the elements are adjoined. The only constraint is that if an element is selected from the coset as a member of the set, its complement cannot be selected since complements are not orthogonal. Thus there are two choices for each of the 2^n elements of the set. That these 2^{2^n} orthogonal sets selected from each coset are distinct is readily apparent. It is equally true that any set selected from different cosets must also be distinct since, given any two cosets, one must have at least one pair, x_i and $x_i + I$, which is not found in the second. Thus there are 2^{2^n} orthogonal sets for each of the 2^{2^n-n-1} cosets of each bi-orthogonal group resulting in $S(2^n)$ orthogonal sets which are equivalent to the orthogonal groups under the complementation and permutation operations defined above. These $S(2^n)$ sets have been obtained from the original group by column permutation, column complementation, row complementation and, finally, row permutation, in that order. All sets obtainable by this sequence of operations have been counted. To state that there are no more orthogonal sets in this equivalence class requires demonstration of the fact that some other order of operations could not have resulted in still another set. By changing from 1, 0 to -1, +1 notation, it is apparent that the complementation and permutation operations may be represented by the matrices C and P where C has only the diagonal elements +1 and -1 and P is a conventional permutation matrix. In addition, there is always a diagonal matrix, C_2, such that $C_1P = PC_2$ for any P and C_1, since

$$C_2 = PC_1P^{-1}$$

and this transformation simply takes c_{ii} into c_{jj}. Thus, if H_1 and H_2 are in the same equivalence class, there exists some set of matrices P_i, C_i such that

$$P_1 \; C_1 \; P_2 \; C_2 \cdots P_m \; C_m \; H_1 \; C_{m+1} \cdots C_n \; P_n = H_2.$$

But since $C_i \; P_i = P_i \; C_j$ and $C_i \; C_j = C_k$ and $P_k \; P_r = P_q$ the above may be written

$$P_a C_a H_1 C_b P_b = H_2 .$$

Again, because P and C commute in the above manner and because matrix multiplication is associative, it follows that the order of permutation and complementation is irrelevant and that all sets can be obtained by the sequence of operations described above.

A short table indicates the order of magnitude of the numbers $S(2^m)$:

m	$S(2^m)$
1	4
2	32
3	122,880
4	$8^+ \times 10^{15}$

Thus, for a still relatively small orthogonal dictionary of size $2^4 = 16$, more than 8×10^{15} dictionaries would need to be investigated to be able to proclaim one of them optimum. This number obviously grows extremely rapidly and very quickly becomes too large for any conceivable computer.

C. The Structure of Orthogonal Groups

In the previous section it was shown that all orthogonal groups are equivalent under permutation of columns to a group whose generators have the form

$$\begin{array}{l} 00\ldots\ldots\ldots\ldots\ldots\ldots\ldots 0 \\ 00\ldots\ldots\ldots 011\ldots\ldots\ldots 1 \\ 00\ldots 1\ldots 10\ldots 01\ldots 1 \end{array} \qquad (3.13)$$

etc.

The complete group of order four is then:

$$\begin{matrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0. \end{matrix}$$

Note that this array is symmetric. As was shown in Chapter 1, a group,B, of order 2^n can be generated from a group, A, of order 2^{n-1} in the following manner:

$$B = \begin{matrix} A & A \\ A & \overline{A} \end{matrix} \qquad (1.24)$$

where $\overline{A}$ consists of the complements of the elements in A. But B is just a symmetric array of symmetric arrays and hence is symmetric. By induction, then, in any group of this form there is a one to one correspondence between rows and columns. As a result any such group contains 2^n-1 columns each with one half ones and one half zeros, and one with all zeros. Since any columns of the group coset may be obtained by complementation and permutation of the group B, and since neither of these operations alters the condition described above, except that perhaps the all zeros column becomes an all ones column, it remains true for all cosets. This property of orthogonal cosets was used in Chapter 2.

The statement in Chapter 2, that any subgroup of order 2^{n-p} contains 2^p columns which consist of all zeros or all ones, follows immediately from the discussion above concerning the number of orthogonal groups. If only n-p generators have been selected, the group may be permuted into the form described there containing 2^{n-p} cells of 2^p elements each, the leading cell containing only zeros. Thus the above statement is true for any coset, since these 2^p columns of all zeros occur in any coset formed from this subgroup as columns of all zeros or all ones.

Chapter 4

BOUNDS ON THE INDEX OF COMMA FREEDOM FOR ORTHOGONAL CODES

A. Upper Bounds

In order to obtain an upper bound on the maximum index of comma freedom, consider the maximum index of comma freedom attainable when a code word is correlated with the cyclic permutations of itself and other code words. Since the set of words formed by cyclic permutations of code words is a subset of the possible out-of-synchronization combinations obtainable, an upper bound on the index of comma freedom considering only this subset is certainly an upper bound for the set as a whole. But if x is to differ from all the cyclic permutations of y in at least p places and no more than N-p places, then all cyclic permutations of x and $\overline{x}$ must differ from all the cyclic permutations of y and $\overline{y}$ in at least p places. To prove this statement let x^r represent the N-tuple resulting from permuting x cyclically r places; e.g., if $x = x_1\ x_2\ x_3\ x_4$, $x^1 = x_4\ x_1\ x_2\ x_3$, $x^2 = x_3\ x_4\ x_1\ x_2$, $x^3 = x_2\ x_3\ x_4\ x_1$. Now suppose $N-p \geq |x + y^t| \geq p$ for all t, but assume $|x^r + y^s| < p$. Then $|x^{r+(N-r)} + y^{s+(N-r)}| = |x^N + y^{s-r+N}| = |x + y^{s-r+N}| < p$. But y^{s-r+N} is just some cyclic permutation of y and thus the last statement is contrary to the hypothesis that $|x + y^t| \geq p$. Now suppose that $|x^r + \overline{y}^s| < p$. Then

$$|x + \overline{y}^{\,s-r+N}| < p$$

$$|x + y^{s-r+N} + I| < p$$

$$N - |x + y^{s-r+N}| < p$$

$$|x + y^{s-r+N}| > N - p$$

which is contrary to the hypothesis that $|x + y^t| \leq N - p$ for all t. The same argument is obviously valid for $|\overline{x}^{\,r} + \overline{y}^{\,s}|$ and $|\overline{x}^{\,r} + y^s|$.

Now consider an orthogonal set of size N. From the above argument, each of the elements of the super-set containing all the cyclic permutations of the elements of this orthogonal set and their complements must differ in at least p places from every other element. The number of elements in this set is evidently $2N(N) = 2N^2$, since there are N cyclic permutations of each of the orthogonal elements and their complements. A well-known result from the study of error-correcting codes(25) states that if M binary vectors of dimension N are to differ from each other in at least p = 2e + 1 components then the following inequality must be satisfied:

$$M \leq \frac{2^N}{1 + \binom{N}{1} + \ldots + \binom{N}{e}}, \quad p = 2e + 1. \tag{4.1}$$

The proof is straightforward. If a vector, which can be obtained from one code word by complementing e or fewer of its components, can be obtained from a second code word by complementing e or fewer of <u>its</u> components, then the two words can differ in no more than 2e components. A necessary condition for a vector to be a code word then is that none of its $N_e = \binom{N}{1} + \binom{N}{2} + \ldots + \binom{N}{e}$ neighboring vectors be either a code word or one of the N_e neighbors of any other code word. That is, no two "spheres" of radius e, with code words as the centers, can intersect. The above inequality follows from this. If no two code vectors are to disagree in fewer than 2d components, a similar inequality can be derived. If, for every code word x, a sphere of radius d is constructed with x as a center, then requiring that no two spheres intersect except, perhaps, at a boundary point, is equivalent to demanding that any two code words are separated by a distance of at least 2d. Note that the possibility that two spheres contain a boundary point in common cannot be excluded. Consider now the maximum number of spheres which can contain the same boundary point. Let z be an equal distance d from the code words x and y. Then $z + y = u$, $z + x = v$, $x + y = u + v$ where $|u| = |v| = d$ and $|x + y| = |u + v| \geqq 2d$. But $2d = |u| + |v| \geqq |u + v|$. Thus we must require that $|u + v| = |u| + |v|$. This is easily seen to be true only if no position occupied by a one in v is also occupied by a one in u. Consequently, any vector x, consisting of N components, can be common to, at most, $[N/d]$ spheres, where the brackets denote "the integral part of". Each code word x then eliminates at least

$$1 + \binom{N}{1} + \ldots + \binom{N}{d-1} + \frac{1}{[N/d]} \binom{N}{d}$$

vectors from consideration. Hence the number of code words is bounded by

$$M \leq \frac{2^N}{1 + \binom{N}{1} + \ldots + \binom{N}{d-1} + \frac{1}{[N/d]} \binom{N}{d}} \tag{4.2}$$

for a code in which the minimum distance $p = 2d$. The following table shows the maximum value of p obtained from the above inequalities for orthogonal codes of length N for several values of N:

Table 4.1

Upper Bounds on the Indices of Comma Freedom

N	p_{max}
4	0
8	2
16	4
32	13
64	30

An independent upper bound on the index of comma freedom for orthogonal codes is readily obtainable from the fact, observed earlier, that the sum of the squares of the N correlations of any sequence with the N code words is constrained:

$$\sum_{i=1}^{N} a_i^2 = 1 \tag{3.6}$$

Now $\max a_i^2 \geq \text{ave } a_i^2 \equiv \alpha^2$. But

$$\sum_{i=1}^{N} a_i^2 = N\alpha^2 = 1$$

$$|\alpha| = \frac{1}{\sqrt{N}} .$$

Therefore,

$$\max |a_i| = \left| \left[\max a_i^2\right]^{\frac{1}{2}} \right| \geq \frac{1}{\sqrt{N}}$$

$$\text{and } \min \max |a_i| = \frac{1}{\sqrt{N}} \quad . \qquad (4.3)$$

The maximum out-of-phase correlation is then at least $\frac{1}{\sqrt{N}}$ in absolute value. Thus $\max_i |x^i \cdot y| = \max_i \left| \frac{N-2D_i}{N} \right| \geq \frac{1}{\sqrt{N}}$ and

$$\max D_i \geq N - \frac{N(1 - \frac{1}{\sqrt{N}})}{2}$$

$$\min D_i \leq \frac{N(1 - \frac{1}{\sqrt{N}})}{2} \quad .$$

This provides a new upper bound on the index of comma freedom:

$$p \leq \frac{N(1 - \frac{1}{\sqrt{N}})}{2} \quad . \qquad (4.4)$$

It may be readily verified that this upper bound on the index of comma freedom, p, is larger than the previous upper bound for $N < 64$. For $N = 64$ the opposite situation is true and continues to be so for all larger values of N.

Some of these bounds can be improved slightly by some additional observations. First note that any upper bound on p obtained by considering only cyclic permutations of the code words must be an even integer. To verify this, let x and y be code words, and again let x_i, the *ith* component of x, be $\frac{1 - 2\,\xi_i}{\sqrt{N}}$, and similarly, let y_i be

$\frac{1 - 2\eta_i}{\sqrt{N}}$ where $\xi_i, \eta_i = 0$ or 1. Then

$$C_k = \frac{N - 2D(k)}{N} = \sum x_i y_{i+k} = \frac{1}{N} \sum (1 - 2\xi_i)(1 - 2\eta_{i+k})$$

$$D(k) = \sum \xi_i + \sum \eta_{i+k} - 2 \sum \xi_i \eta_{i+k} .$$

Since $\sum_{i=1}^{N} \eta_{i+k} = \sum_{i=1}^{N} \eta_i$ and since D(0) is an even number for two orthogonal sequences of even length greater than 2, then $\sum \xi_i + \sum \eta_i$ is an even number. Hence D(k) is even for all k. Note that this is also true if x = y. Since all of the bounds above for $N < 64$ are obtained by considering only cyclic shifts, these bounds, if odd, can be replaced by the next lower even number.

For the cases $N \gtrsim 64$ the best upper bound thus far states that $p \leq \frac{N(1 - \frac{1}{\sqrt{N}})}{2}$. This implies that each word in the code forms a "perfect" sequence in the sense that its correlation with all its phase shifts is $\pm \frac{1}{\sqrt{N}}$. Thus

$$\sum_{k=1}^{N} C_k = 1 + \frac{1}{\sqrt{N}} \left[\upsilon - (N - 1 - \upsilon) \right] = \sum_{k=1}^{N} \sum_{i=1}^{N} x_i x_{i+k} \tag{4.5}$$

$$= \left(\sum_{i=1}^{N} x_i \right)^2 = \frac{(N - 2p)^2}{N}$$

where p is the number of $-\frac{1}{\sqrt{N}}$'s (1's) in the code word x, and υ is the number of times the correlation $+\frac{1}{\sqrt{N}}$ occurs. Hence:

$$\begin{aligned} (N - 2p)^2 &= N + \sqrt{N}\left[2\nu + 1 - N\right] \\ N + N\sqrt{N} - 4p\sqrt{N} + \frac{4p^2}{\sqrt{N}} - \sqrt{N} &= 2\nu + 1. \end{aligned} \tag{4.6}$$

In order that $\sqrt{N}$ be an integer we require that $N = 2^{2m}$. Thus if $\sqrt{N}$ is an integer, it is an even integer. In the above equation, then, all terms on the left side except possibly $\frac{4p^2}{\sqrt{N}}$ are even, while the right side is odd. For a solution to exist, $\frac{4p^2}{\sqrt{N}}$ must be odd. If N is of the form $N = 2^{2(2m+1)}$, no solution exists since:

$$\frac{4p^2}{\sqrt{N}} = \frac{4p^2}{2^{2m+1}} = \tfrac{1}{2}\left(\frac{p}{2^{m-1}}\right)^2 = \tfrac{1}{2}\,q^2.$$

But $\frac{1}{2}q^2$ can never be an odd integer (since, if $q^2 = 2s$, q must be even, $q = 2t$; $q^2 = 4t^2$, and $\frac{q^2}{2} = 2t^2$). For the cases $N = 2^{2(2m+1)}$ the upper bound on p can thus be reduced by 2, since no "perfect" sequences of these lengths exist and since the number of disagreements between any sequence and a cyclic permutation of itself is an even integer.

Finally, it will be shown in Chapter 7 that no orthogonal comma-free group coset dictionary exists for the case N = 8. This does not exclude the possibility, however, that an orthogonal set obtained by complementing some of the rows of a coset may be comma-free for N = 8.

B. Lower Bounds

In this section we show an iterative procedure for construction of comma-free bi-orthogonal codes. This establishes a constructive lower bound on comma-free codes for the bi-orthogonal (and, trivially, the orthogonal) case.

Consider the following construction:

$$C = A \times B \cup \overline{A \times B} \tag{4.7}$$

where x denotes the "Kronecker product" defined as follows:

$$A \times B = (a_{ij}) \times (b_{km})$$

$$= \begin{pmatrix} a_{11}B & a_{12}B & \cdots\cdots & a_{1M}B \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ a_{M1}B & \cdots & \cdots\cdots & a_{MM}B \end{pmatrix} \tag{4.8}$$

A is an M x M matrix and B is an N x N matrix. Here a_{ij} and b_{km} are either +1 or -1. Thus C is the union of two MN x MN matrices, the second being obtained from the first by multiplying every component by -1. Now suppose A and B are orthogonal code dictionaries. Consider the correlation between two arbitrary rows of A x B, viz

$$a_{i1}\beta_k,\ a_{i2}\beta_k,\ \ldots\ldots,\ a_{iM}\beta_k$$
$$a_{j1}\beta_m,\ a_{j2}\beta_m,\ \ldots\ldots,\ a_{jM}\beta_m$$

where β_υ represents the υth row of the matrix, B. Then, if $m \neq k$, these two rows are certainly orthogonal due to the orthogonality of the rows of B. Further, since the above rows can clearly be permuted to the form

$$b_{k1}\,\alpha_i,\ b_{k2}\,\alpha_i,\ \ldots\ldots,\ b_{kM}\,\alpha_i$$
$$b_{m1}\,\alpha_j,\ b_{m2}\,\alpha_j,\ \ldots\ldots,\ b_{mM}\,\alpha_j$$

where α_j denotes the jth row of A, and since a permutation of the columns of a matrix does not alter the correlations between its rows; the rows of A x B are also orthogonal if $i \neq j$, due to the orthogonality of the rows of A. But as inspection of equation 4.8 quickly reveals, there are no two distinct rows of A x B for which both $i = j$ and $m = k$. Hence all rows of A x B are mutually orthogonal.

Let $A \cup \overline{A}$ be a bi-orthogonal code with an index of comma freedom p_M and $B \cup \overline{B}$ be one with an index p_N. The construction, C, is then a bi-orthogonal code in which the following represents a typical word:

$$X = x\,x\,\overline{x}\,\overline{x}\,\ldots\ldots\,x\,\overline{x}$$

where x, a row of B, contains N terms and there are a total of M x's and $\overline{x}$'s.

We will consider the out-of-phase correlations by investigating the following four situations:

(a) If the phase position $k \neq qN$ ($q = 1, 2, \ldots, M-1$) then each x in X agrees with the corresponding out-of-phase symbols in not more than p_N and not less than $N-p_N$ places. This follows from the fact that the x's are code words of B and that all possible out-of-phase sequences of length N to which x or $\bar{x}$ can be compared here are included in the set of overlaps generated by the code B. Thus $p_{MN} \geq Mp_N$ for this case.

(b) $k = qN$ ($q = 1, 2, \ldots, M-1$) and X is compared to overlaps involving only non-x and $\bar{x}$ terms. Then, since the vectors in B are orthogonal, $p_{MN} = MN/2$.

(c) $k = qN$ ($q = 1, 2, \ldots, M-1$) and the out-of-phase vector involves only x components. Then, trivially, $p_{MN} = Np_M$ since every disagreement in A implies N disagreements in C.

(d) $k = qN$ ($q = 1, 2, \ldots, M-1$) and the out-of-phase vector contains both x, $\bar{x}$ and non-x, $\bar{x}$ terms. If the first part of the sequence to which X is compared contains x's, then every disagreement in A implies N disagreements in C. In the second part of the sequence, containing υ elements of A, there are υ N/2 disagreements due to orthogonality.

If the number of disagreements, d, in the corresponding comparison in A is equal to $d_1 + d_2$ where d_1 is the number of disagreements in the first $M-\upsilon$ places and d_2 the number in the last υ places, then

$$M - p_M \geq d_1 + d_2 \geq p_M.$$

But in the bi-orthogonal codes, if either word of the sequence consisting of the overlap of two words to which a code word is compared is complemented another possible comparison results. Thus:

$$M - p_M \geq d_1 + \upsilon - d_2 \geq p_M$$

$$M - p_M - \upsilon \geq d_1 - d_2 \geq p_{M-\upsilon}.$$

Combining this last equation with the equation above, one obtains:

$$\frac{2M - \upsilon - 2p_M}{2} \geq d_1 \geq \frac{2p_M - \upsilon}{2} . \tag{4.9}$$

Hence, since

$$d_{MN} = N(d_1 + \frac{\upsilon}{2})$$

then

$$N(M - p_M) \geq d_{MN} \geq Np_M$$

and thus

$$p_{MN} \geq Np_M \quad . \tag{4.10}$$

The same result obviously follows if the second part rather than the first part of the sequence contained the x and $\overline{x}$ terms. Note that cases (b) and (c) are just special cases of this last case (d).

Combining the results from these four cases, one finds that

$$p_{MN} \geq \min (Np_M, Mp_N). \tag{4.11}$$

If M = N

$$p_{N^2} \geq Np_N \quad .$$

The index of comma freedom then grows at least as rapidly as the dictionary size.

Note that to establish lower bounds for all bi-orthogonal dictionaries of order 2^N it must be shown that there are comma-free codes for the cases N = 16, 32, 64 and 128, since none exist for the case N = 8 (see Chapter 7). This follows from the fact that no product of the above integers yields a dictionary of less than $2^8 = 256$ words, and thus no smaller comma-free dictionary can be formed in the manner described. To see that those numbers are sufficient, note that $2^n = 2^4 \times 2^{n-4}$ and that, if comma-free codes exist for n = 4, 5, 6, 7, they can be obtained for all larger n by recursion. The improved upper bounds on the index of comma freedom, p_u, and the index p', of the best code yet attained are listed in the following table. These latter values were obtained by selecting by various heuristic methods a coset-leader for a group dictionary and determining the resulting comma freedom on a computer.

Table 4.2

Lower and Improved Upper Bounds
on the Indices of Comma Freedom

N	p_u	p'
8	2 (0)	0
16	4	2
32	12	6
64	26	14
128	58	34

Before leaving this subject, consider the following estimate of the index of comma freedom obtainable with large code dictionaries, which, although admittedly heuristic in derivation is nonetheless somewhat illuminating. The method is as follows: Choose an orthogonal group of order N and construct a coset leader by selecting N consecutive binary symbols at random, with $p(0) = p(1) = \frac{1}{2}$. Now consider the following orthogonal coset:

$$H = G + c + c^k + y \tag{4.12}$$

where c is the random coset leader, c^k is the sequence c shifted cyclically k positions to the left, and y is a sequence consisting of the last N-k symbols of one of the group elements followed by the first k symbols of a second group element. G + y is an orthogonal group coset, and $c + c^k$ is a coset leader whose symbols form a random binary sequence. Hence, the elementsof H are also random binary sequences. But note that, if $-\frac{1}{N}$'s are substituted for the 1's in H and $\frac{1}{N}$'s for the 0's, the sum of the symbols forming the ith element, h^i, of H is just the correlation between the ith word of the dictionary G + c and a possible overlap occurring at a phase position k. Since the sequence$\left\{h_j^i\right\}$is a random sequence with $p(-\frac{1}{N}) = p(\frac{1}{N}) = \frac{1}{2}$, the sum $S_i^2 = (\sum_{j=1}^{N} h_j^i)^2$ is a random variable with a mean

$$E(S_i^2) = E\sum_{j=1}^{N}\sum_{k=1}^{N} h_j^i\, h_k^i = \sum_{j=1}^{N}\sum_{k=1}^{N} E(h_j^i\, h_k^i) = N^{\frac{1}{2}}\sum_{j=1}^{N}\sum_{k=1}^{N}\delta_{jk} = \frac{1}{N}$$

and $E(\ (S_i^2)^2\ ) - (E\ (S_i{}^2)\ )^2 = \sigma^2(S_i{}^2)$ (4.13a)

$$= E\sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{m=1}^{N}\sum_{n=1}^{N} h_j^i\, h_k^i\, h_m^i\, h_n^i - \frac{1}{N^2}$$

$$= \frac{1}{N^4}\sum_{\substack{j=1\\ j\neq m}}^{N}\sum_{k=1}^{N}\sum_{m=1}^{N}\sum_{N=1}^{N}\delta_{jk}\ \delta_{mn} \tag{4.13b}$$

$$+\frac{1}{N^4}\sum_{\substack{j=1\\ j\neq n}}^{N}\sum_{k=1}^{N}\sum_{m=1}^{N}\sum_{n=1}^{N}\delta_{jm}\ \delta_{kn}$$

$$+\frac{1}{N^4}\sum_{\substack{j=1\\ j\neq k}}^{N}\sum_{k=1}^{N}\sum_{m=1}^{N}\sum_{n=1}^{N}\delta_{jn}\ \delta_{km}$$

$$+\frac{1}{N^4}\sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{m=1}^{N}\sum_{n=1}^{N}\delta_{jk}\ \delta_{km}\ \delta_{mn} - \frac{1}{N^2}$$

$$= \frac{3N(N-1) + N - N^2}{N^4} = \frac{2(N^2 - N)}{N^4} \approx \frac{2}{N^2}\ .$$

Thus the expected value of the square of the correlation between any word and any out-of-phase sequence is $\frac{1}{N}$ with a variance that rapidly approaches zero for large N. This suggests that a group code with a random coset leader tends to approach asymptotically the maximum attainable index of comma freedom, namely that corresponding to a correlation of $\pm \frac{1}{\sqrt{N}}$ between any word and any out-of-phase sequence. It is interesting to observe, in this regard, that, even for small values of N, the best codes that have been found have generally included those with some cyclic permutation of a pseudo-random sequence[26] as a coset leader.

Chapter 5

SYNCHRONIZATION WITH COMMA-FREE CODES

A. The Probability of Correct Synchronization

Several factors which somewhat complicate the calculation of the time necessary to obtain synchronization to the desired accuracy should be observed. First, the correct phase position needs to be determined only once in any uninterrupted sequence, and since this knowledge reduces the number of correlation variables to be considered by a factor of N, it is clearly more efficient to estimate the correct phase initially, before attempting to decode. Thus the optimum synchronization method need not attempt to discern the received words at all, but only the correct phase position. Second, the random variables representing the outputs of the correlators at difference instants of time are not necessarily independent, as will be demonstrated later. Finally, except in the case of the effectively unattainable "perfect" comma-free codes, in which all the random variables have means whose absolute value is $\frac{1}{\sqrt{N}}$, the expected value of the correlator outputs in the out-of-phase position is unknown, the only restrictions being on the sum of the squares of these means for any given phase position, and an upper bound on their absolute magnitude.* Yet, it is desirable to state that, regardless of the sequence of received words, synchronization can be obtained in τ seconds with a given probability, P_s. The following technique for exploiting the self-synchronizing properties of

* Although a maximum likelihood phase detector can be determined, it is virtually impossible to obtain meaningful synchronization time bounds with it, due, in particular, to this last situation. The upper bound obtained by assuming that all means are at the maximum possible value is absurdly large.

comma-free codes is evidently sub-optimum. Its justification rests, first of all, in the results it achieves and, secondly, in the fact that an upper bound on the synchronization time using this method is computable.

The scheme is as follows:

Consider the outputs of the N correlators at some given phase position, k. These are mutually independent* Gaussian random variables with one (normalized) mean at one and the rest at zero in the in-phase position, and all the means less than some value, $\frac{N-2p}{N}$, in the out-of-phase position. Now define a threshold and observe whether it has been exceeded in absolute value by any of the random variables in question. The probability that the threshold is exceeded at a given phase position, k, is given by

$$p_k = 1 - \prod_{i=1}^{N} P_i \qquad (5.1)$$

where

$$P_i = P_N(\mu_i) = \frac{1}{(2\pi)^{\frac{1}{2}}} \int_{\frac{-\alpha - \mu_i}{\sigma}}^{\frac{\alpha - \mu_i}{\sigma}} e^{-\frac{t^2}{2}} dt \qquad (5.2)$$

* The independence of these variables is a consequence of the orthogonality of the code words. Thus, if

$$Z_i = \int_o^T x_i(t)\left[y(t) + n(t)\right] dt \quad \text{and} \quad Z_j = \int_o^T x_j(t)\left[y(t) + n(t)\right] dt$$

then

$$E(Z_i Z_j) = \left[\int_o^T x_i(t)y(t)dt\right]\left[\int_o^T x_j(t)y(t)dt\right] + \int_o^T x_i(t)x_j(u)\delta(t-u)\, dtdu$$

$$= \left[\int_o^T x_i(t)y(t)dt\right]\left[\int_o^T x_j(t)y(t)dt\right] = E(Z_i)\, E(Z_j).$$

and where α represents the threshold, and μ_i is the mean, and $\sigma = \sigma_n T^{\frac{1}{2}}$ the standard deviation of the Gaussian variable corresponding to the output of the ith correlator. The distribution of the number of times the threshold is exceeded in m trials (for a given phase position k) is simply the binomial distribution with the probability of a success per trial, p_k, as defined above. It is desired to estimate the number of trials necessary to assure, with a pre-assigned probability, that the number of successes in the in-phase position is greater than the number of successes for any other phase.

Consider now the correlation of the outputs of two different correlators at different instants of time:

$$E(c_{ik}\, c_{jm}) - E(c_{ik})\, E(c_{jm})$$

$$= E \int_0^T \int_0^T x_i(t)\left[y(t + \frac{kT}{N}) + n(t + \frac{kT}{N})\right] x_j(u)\left[y(u + \frac{mT}{N}) + n(u + \frac{mT}{N})\right] dtdu$$

$$- E(c_{ik})\, E(c_{jm})$$

$$= \sigma_n{}^2 \int_0^T \int_0^T x_i(t)\, x_j(u) \;\; \delta(t - u + \frac{(k-m)}{N} T)\; dtdu$$

$$= \sigma_n{}^2 \int_0^{T(1 - \frac{(k-m)}{N})} x_i(t)\, x_j(t + \frac{(k-m)}{N} T)\; dt, \qquad (k-m) \leq N \qquad (5.3)$$

Hence, the correlation between the random variables corresponding to different phase shifts is not, in general, zero. The variables in this case then are not statistically independent. As in Chapter 2, however, a lower bound on the probability of identifying the correct position is given by the following equation:

$$P_s \geq P_B(m, r, p_o) - \left[P_B(m, r, p_1) + \dots + P_B(m, r, p_{N-1})\right] \quad (5.4)$$

where the function $P_B(m, r, p_i)$ represents the probability of r or more successes in m trials when the probability of an individual success is p_i. The probability $P_B(m, r, p_o)$ for the in-phase situation may be readily calculated for a given threshold from the above expression for p_o. But the out-of-phase probability depends upon the position of the means. An upper bound for the probability $P_B(m, r, p_i)$ which is independent of the distribution of the correlation means would be desirable. Since $P_B(m, r, p) = \sum_{i=r+1}^{n} \binom{n}{i} p^i q^{n-i}$ increases monotonically as p increases, the above upper bound is attained for that distribution of the means for which p is maximized. Since $P_N(\mu_i) = P_N(-\mu_i)$, it can be assumed without loss of generality that all the means are positive. Further, from previous considerations it is known that $\sum \mu_i^2 = A^2T^2$ and $\left|\mu_i\right| \leq \left(\frac{N-2p}{N}\right)AT$, where the means have not been normalized as before. The variance, of course, is the same for all distributions. Consider now an arbitrary arrangement of the means, μ_i, subject to the above conditions. Let us select two of these means, increase one and decrease the other in such a way that the sum of the squares is held fixed, and investigate the behavior of the probability $p = 1 - \prod_i P(\mu_i)$. Let

$$\sum \mu_i^{\,2} = A^2T^2, \quad \mu_j^{\,2} + \mu_k^{\,2} = K^2, \qquad \mu_k \geq \mu_j$$

then

$$\begin{aligned} \operatorname{sgn}\left\{\frac{\partial p}{\partial \mu_k}\right\} &= \operatorname{sgn}\left\{\left(-P_j \frac{\partial P_k}{\partial \mu_k} - P_k \frac{\partial P_j}{\partial \mu_k}\right) \prod_{i\neq k,j} P(\mu_i)\right\} \\ &= \operatorname{sgn}\left\{-\frac{1}{\mu_k}\,\frac{1}{P_k}\,\frac{\partial P_k}{\partial \mu_k} + \frac{1}{\mu_j}\,\frac{1}{P_j}\,\frac{\partial P_j}{\partial \mu_j}\right\} . \end{aligned} \tag{5.5}$$

The last expression was obtained by observing that $\dfrac{\partial P_j}{\partial \mu_k} = -\dfrac{\mu_k}{\mu_j}\,\dfrac{\partial P_j}{\partial \mu_j}$ and by dividing both terms by $\mu_k \prod_i P_i$, a positive quantity. Now consider $\operatorname{sgn}\left\{\dfrac{\partial}{\partial \mu}\left(-\dfrac{1}{\mu}\,\dfrac{1}{P}\,\dfrac{\partial P}{\partial \mu}\right)\right\}$ where $P = P_N(\mu)$. If this equation is always positive (or negative) over the range of μ of interest, the p is monotonically increasing (or decreasing) as a function of k over this range. But:

$$\begin{aligned} &\operatorname{sgn}\left\{\frac{\partial}{\partial \mu}\left(-\frac{1}{\mu}\,\frac{1}{P}\,\frac{\partial P}{\partial \mu}\right)\right\} \\ &= \operatorname{sgn}\left\{\left(\frac{1}{\mu^2}\,\frac{\partial P}{\partial \mu} - \frac{1}{\mu}\,\frac{\partial^2 P}{\partial \mu^2}\right)\frac{1}{P} + \frac{1}{\mu P^2}\left(\frac{\partial P}{\partial \mu}\right)^2\right\} . \end{aligned} \tag{5.6}$$

Since the last term is always positive, the above expression is positive at least over the range of μ for which the term in the first set of parentheses is positive. Now:

$$\operatorname{sgn}\left\{\frac{1}{\mu^2}\,\frac{\partial P}{\partial \mu} - \frac{1}{\mu}\,\frac{\partial^2 P}{\partial \mu^2}\right\}$$

$$= \operatorname{sgn}\left\{\frac{1}{\mu^2}\left(e^{-\frac{(\alpha+\mu)^2}{2\sigma^2}} - e^{-\frac{(\alpha-\mu)^2}{2\sigma^2}}\right)\right.$$

$$\left. + \frac{1}{\mu}\left(\frac{(\alpha+\mu)}{\sigma^2}\, e^{-\frac{(\alpha+\mu)^2}{2\sigma^2}} + \frac{(\alpha-\mu)}{\sigma^2}\, e^{-\frac{(\alpha-\mu)^2}{2\sigma^2}}\right)\right\}$$

$$= \operatorname{sgn}\left\{\beta x \cosh \beta x - (1 + x^2) \sinh \beta x\right\} \qquad (5.7)$$

where $\beta = \frac{\alpha}{\sigma}$, $x = \frac{\mu}{\sigma}$. Expanding the hyperbolic functions in power series (which converge for all x) one obtains:

$$\operatorname{sgn}\left\{\frac{1}{\mu^2}\,\frac{\partial P}{\partial \mu} - \frac{1}{\mu}\,\frac{\partial^2 P}{\partial \mu^2}\right\} = \operatorname{sgn}\left\{\sum_{n=1}^{\infty} \frac{\beta^{2n-1}}{(2n-1)!}\left(\frac{\beta^2}{2n+1} - 1\right)x^{2n+1}\right\} \qquad (5.8).$$

This last expression provides all the necessary information concerning the conditions for which the function under consideration is monotonically increasing. In particular, if $\beta^2 = \frac{\alpha^2}{\sigma^2} \leq 3$, then the sign of the above expression is always negative since all terms of the series are negative. If $2k+1 < \beta^2 < 2k+3$, the first k terms in the series are positive; the remaining terms all being negative. Since the series is positive for small values of x and negative for large values of x, the function must cross the x axis for some positive value of x. Let $x = x_o$ represent the first zero. Then

$$\sum_{n=1}^{k} a_n x_o^{2n+1} - \sum_{n=k+1}^{\infty} b_n x_o^{2n+1} = 0; \qquad a_n, b_n > 0.$$

Now let $x = \gamma x_o$, $\gamma > 1$. Then:

$$\sum_{n=1}^{k} a_n(\gamma x_o)^{2n+1} - \sum_{n=k+1}^{\infty} b_n(\gamma x_o)^{2n+1}$$

$$< \gamma^{2k+1}\left[\left(\sum_{n=1}^{k} a_n x_o^{2n+1} - \sum_{n=k+1}^{\infty} b_n x_o^{2n+1}\right) - (\gamma^2-1)\sum_{n=k+1}^{\infty} b_n x_o^{2n+1}\right]$$

$$< -(\gamma^2 - 1)\sum_{n=k+1}^{\infty} b_n x_o^{2n+1} < 0 \text{ for all } \gamma > 1. \qquad (5.9)$$

Thus there are no more zeros to the right of x_o which was by hypothesis the first zero to the right of the origin. To recapitulate, for $\frac{\alpha}{\sigma} > \sqrt{3}$, the function has one and only one zero and is positive for all $x = \frac{\mu}{\sigma}$ less than the value of $x = x_o$ at that zero. The value of this zero is readily calculated numerically and is tabulated in the following table as a function of β :

Table 5.1

The Zeros of the Function $f(\beta,x) = \beta x \cosh\beta x - (1 + x)^2 \sinh\beta x$

β	x_o
2	1.200
2.5	2.00
3	2.618
3.5	3.061
4	3.735
5	4.795

If the threshold $\frac{\alpha}{\sigma} > \sqrt{3}$ and $x_o \sigma > (\frac{N-2p}{N}) AT = \max(\mu)$, then p increases as $\mu_k > \mu_j$ is increased subject to the constraint that the sum $\mu_k^2 + \mu_j^2 = K^2$. Consider all possible arrays of means for the out-of-phase distributions. Since p increases as the larger of any pair of means is increased while the smaller is decreased so as to keep the sum of the squares constant, it is apparent that the maximum value of p is attained when as many means as possible are at the maximum allowable value while the rest are at zero. If k is the number of means at the maximum value,

$\frac{1}{A^2 T^2} \sum \mu_i^2 = k(\frac{N-2p}{N})^2 = 1$ implies that $k = \frac{N^2}{(N-2p)^2}$. If this is not an integer then the maximum value of p is clearly attained for

$k = \left[\frac{N^2}{(N-2p)^2} \right]$ where the brackets denote the "integer part of". There are then N-k-1 means at zero with the remaining mean at some intermediate value in order to satisfy the constraint that $\sum \mu_i^2 = A^2T^2$. Thus, for a given signal-to-noise ratio and given threshold, an upper bound on p is obtained, so long as the threshold is such that $x = x_o$ is greater than $\frac{\mu_{max}}{\sigma}$ where μ_{max} is the maximum possible out-of-phase mean.

Under these conditions, upper bounds on the synchronization times can be calculated. Some of these results are summarized in Table 5.2. Again, the probability of a bit error in synchronous operation is assumed to be .001, and the number of words, m and m', necessary to obtain correct synchronization with a probability of .999 and of .9999, respectively, is determined. As in Chapter 2, the rate of transmission is assumed to be 45 bits per second. The probability of correct synchronization is estimated by

$$P_s \geq P_B(m, r, p_o) - (N-1)\, P_B(m, r, p_1) \qquad (5.4a)$$

and the values of the $P_B(m, r, p_j)$ terms are taken from a table of the cumulative binomial probability distribution(27) using the bounds for the probabilities p_j, as determined above. These latter values are included in Table 5.2. The values of the function $P_N(\mu_i)$ of equation 5.2, necessary for the calculation of these probabilities are, of course, also obtained from a set of tables.(28)

Table 5.2

Upper Bounds on the Synchronization Times for Comma-Free Codes

N	8	16	32	64	128
p	0	2	6	14	34
γ	-	.95	.95	.9375	.925
δ	-	.906	.906	.904	.886
R	4	3.25	2.88	2.50	2.40
$m(.999)$	-	72	36	28	18
r	-	30	13	10	6
mT(secs.)	-	6.4	4.0	3.73	2.80
$m'(.9999)$	-	110	43	34	21
r'	-	46	14	11	6
$m'T$(secs.)	-	9.78	4.78	4.53	3.27
p_o	-	.6010	.6056	.6338	.6667
p_1	-	.2134	.0796	.0587	.0168

The threshold $\gamma = \frac{\alpha}{AT} = \frac{\beta\,\sigma}{AT} = \frac{\beta\,\sigma_n}{AT^{\frac{1}{2}}}$ that was used is included in the above table; it is not necessarily the optimum value. The quantity $\delta = \frac{\mu_o}{AT} = \frac{x_o\sigma_n}{AT^{\frac{1}{2}}}$ corresponding to the zero x_o is also given, and it is observed that the threshold is always large enough so that the above argument concerning the maximum of p is valid. That is, if $\delta = \frac{x_o\sigma_n}{AT^{\frac{1}{2}}} > (\frac{N-2p}{N})$, then $x_o > \frac{\mu_{max}}{\sigma}$. It is easily seen that the necessary inequality remains valid for all larger signal-to-noise ratios if δ remains fixed. This follows from the fact that x_o grows at least as rapidly as β, and that $\beta = \gamma\,(\frac{ST_b}{N_o})^{\frac{1}{2}}(2\log_2 N)^{\frac{1}{2}}$ grows as rapidly with $(\frac{ST_b}{N_o})$ as $\frac{\mu_{max}}{\sigma_n T^{\frac{1}{2}}} = (\frac{N-2p}{N})(\frac{ST_b}{N_o})^{\frac{1}{2}}(2\log_2 N)^{\frac{1}{2}}$.

In addition, it follows immediately that if $\mu_{max} < x_o\sigma$, the synchronization time decreases as $|\mu_{max}|$ is decreased as was contended in Chapter 3. If $\mu_{max} > x_o\sigma$ this is no longer necessarily true, but at these values of μ the probability $p(k\neq 0)$ is very nearly equal to $p(k=0)$ (cf. equation 5.1), and the synchronization time is large regardless of the value of μ_{max}.

For the case N = 16, the time necessary for synchronization with these codes is greater than in the random source situation. This process, of course, has the advantage that no assumptions have been made concerning the randomness of the transmitted sequence. As N increases the synchronization actually decreases; when N = 128, there is a factor of 16 improvement over the random source scheme.

B. Asymptotic Results

It is of interest to investigate the asymptotic behavior of the synchronization time as the number of code words, N, becomes large. From equation 5.2 it is seen that a lower bound on the probability that the threshold is exceeded in the in-phase position is given by:

$$p_o = 1 - \prod_{i=1}^{N} P_i > 1 - \frac{1}{(2\pi)^{\frac{1}{2}}} \int_{-(1+\gamma)\frac{\mu}{\sigma}}^{-(1-\gamma)\frac{\mu}{\sigma}} e^{-\frac{t^2}{2}} dt \qquad (5.10)$$

while the out-of-phase probability is bounded by:

$$p_1 \leq 1 - \left[\frac{1}{(2\pi)^{\frac{1}{2}}} \int_{-(\gamma+\lambda)\frac{\mu}{\sigma}}^{(\gamma-\lambda)\frac{\mu}{\sigma}} e^{-\frac{t^2}{2}} dt\right]^N \qquad (5.11)$$

where $\mu = AT$ and $\rho = \lambda\mu$ is the maximum possible correlation between an out-of-phase sequence and a code word.

Note that, since $\frac{\mu}{\sigma} = \left(\frac{ST_b}{N_o}\right)^{\frac{1}{2}} (2\log_2 N)^{\frac{1}{2}} \geq (2\log_e N)^{\frac{1}{2}}$, p_o asymptotically approaches one for $\gamma < 1$. Since

$$P_s \geq P_B(m, r, p_o) - (N-1)\, P_B(m, r, p_1)$$

$$\geq P_B(m, 1, p_o) - (N-1)\left[1 - (1-p_1)^m\right] \qquad (5.12)$$

$$\geq P_B(m, 1, p_o) - (N-1)\, m\, p_1$$

when p_o and p_1 are replaced by the bounds given above, and since the first term asymptotically approaches one for any value of $m \geq 1$, it suffices to show that $N\, p_1$ approaches zero to establish that correct synchronization is available asymptotically, as soon as a complete word has been received.

But:

$$N\,p_1 < N\left\{1 - \left[1 - \frac{2}{(2\pi)^{\frac{1}{2}}}\int_{(\gamma-\lambda)\frac{\mu}{\sigma}}^{\infty} e^{-\frac{t^2}{2}}\,dt\right]^N\right\}$$

$$\leq \left(\frac{2}{\pi}\right)^{\frac{1}{2}} N^2 \int_{(\gamma-\lambda)\frac{\mu}{\sigma}}^{\infty} e^{-\frac{t^2}{2}}\,dt$$

$$\leq \left(\frac{2}{\pi}\right)^{\frac{1}{2}} N^2 \frac{e^{-\frac{(\gamma-\lambda)^2}{2}\left(\frac{\mu}{\sigma}\right)^2}}{(\gamma-\lambda)\frac{\mu}{\sigma}} \qquad (5.13)$$

$$= \left(\frac{2}{\pi}\right)^{\frac{1}{2}} \frac{e^{-\log_2 N\left[(\gamma-\lambda)^2\frac{ST_b}{N_o} - 2\log_e 2\right]}}{(\gamma-\lambda)\left(\frac{ST_b}{N_o}\right)^{\frac{1}{2}}(2\log_2 N)^{\frac{1}{2}}} \;.$$

Thus, if $\frac{ST_b}{N_o} \geq \frac{2\log_e 2}{(\gamma-\lambda)^2}$, $N\,p_1$ does in fact approach zero asymptotically with N. Since, presumably, λ approaches zero as N becomes large, and since γ can be nearly one, this requirement on the signal-to-noise ratio is the same order of magnitude as that necessary to assure that the synchronous error probability approaches zero as N increases.

It will be remembered, in contrast, that the value of m necessary in the random source synchronization technique increased asymptotically more rapidly than $\frac{N}{\log_2 N}$. In addition, of course, that method assumed a random sequence of received words, whereas the comma-free method gives an upper bound over all possible received sequences.

C. Some Comments Concerning the Complexity of the Decoder

The disadvantage of this synchronization technique lies in the increased amount of equipment necessary to decode; the amount of encoding equipment in the satellite or probe, it should be noted, remains essentially the same. This increase in the complexity of the decoding equipment can be diminished or eliminated at the expense of synchronization time by not investigating all of the phase positions simultaneously. Suppose that the N possible phase positions are divided into S blocks of $\frac{N}{S}$ positions each. The $\frac{N}{S}$ positions of each block are to be investigated simultaneously, as before, but the blocks are to be investigated serially. Since the same threshold can be used in both cases, the time necessary to look at a particular phase position is essentially the same regardless of how many other phases are being simultaneously investigated. But since there are S different blocks to be searched sequentially, and each block contains the correct phase position with the probability $\frac{1}{S}$, the expected number of blocks, K, that need to be observed before the correct one is found, is just:

$$E(K) = \sum_{i=1}^{S} \frac{i}{S} = \frac{S(S+1)}{2S} = \frac{S+1}{2} . \qquad (5.14)$$

Thus the increase in the expected search time necessitated by simultaneously investigating only $\frac{N}{S}$ rather than N phase positions is given by the factor $\frac{S+1}{2}$.

This increase can be reduced somewhat by a two-step search.[(29)] The first step consists of investigating each block for a much shorter time than that necessary for synchronization. The resulting information is then used to rank the blocks in decreasing order of the *a posteriori* probability that they contain the correct phase position. A second search then investigates these blocks in order of their rank until the correct position is determined. By a judicious choice of the time spent on the first investigation, the necessary search time can be reduced to approximately one-half that necessitated by a one-step process if S is greater than 5.

This, then, is one method by which the complexity of the equipment necessary for synchronization can be decreased by increasing the synchronization time. Other methods, including, perhaps, a preliminary search using the random source technique, might be applicable in certain practical situations.

Chapter 6

SYNCHRONIZATION OVER THE BINARY SYMMETRIC CHANNEL

A. The Binary Symmetric Channel

Binary codes, in general, and orthogonal and bi-orthogonal codes, in particular, were originally conceived as attempts to achieve a lower error probability over the binary symmetric channel. This channel is so named because each transmitted symbol may be one of two possibilities, say 0 or 1, and the noise and signal energy are such that a zero is as likely to be mistaken for a one as the converse, the probability of this happening being designated by p. Shannon has shown (30) that the channel capacity in this case is given by:

$$C_b = 1 + p \log p + (1-p) \log(1-p) \text{ bits/symbol.} \tag{6.1}$$

It is of interest to compare this capacity with that of the continuous channel. The bandwidth occupancy of the binary channel is defined to be $\frac{1}{2T_s}$, as before (cf. Chapter 1), where T_s is the time per symbol. Now suppose that the two channels are perturbed by white Gaussian noise and that a maximum likelihood detector is used in both cases. Then the capacity of the continuous channel is given by

$$C_c = W \log(1 + \frac{S}{N_o W}) = \frac{1}{2T_s} \log_2(1 + \frac{2ST_s}{N_o}) \text{ bits/second} \tag{6.2}$$

where N_o is the noise power per unit bandwidth. The probability of mistaking a one for a zero (or a zero for a one), p, is given by

$$p = \left(\frac{2}{\pi}\right)^{\frac{1}{2}} \frac{1}{N_o^{\frac{1}{2}} T_s^{\frac{1}{2}}} \int_{-\infty}^{0} e^{-\frac{(x - S^{\frac{1}{2}}T_s)^2}{N_oT_s}} dx$$

$$= \frac{1}{(2\pi)^{\frac{1}{2}}} \int_{-\infty}^{-\left(\frac{2ST_s}{N_o}\right)^{\frac{1}{2}}} e^{-\frac{t^2}{2}} dt \qquad (6.3)$$

$$= \tfrac{1}{2} - \operatorname{erf}\left(\left(\frac{2ST_s}{N_o}\right)^{\frac{1}{2}}\right)$$

since the maximum likelihood symbol detector integrates over one symbol time and proclaims the signal one if the result is positive and zero if not. The expected output if a one is sent is $AT_s = S^{\frac{1}{2}}T_s$ (see Chapter 1), and the variance σ^2 is $\frac{N_oT_s}{2}$.

The limits as $T_s \longrightarrow 0$ of C_c and C_b are easily obtained (cf. equation 1.19):

$$\lim_{T_s \to 0} C_c = \lim_{T_s \to 0} \frac{1}{2T_s} \log_2\left(1 + \frac{2ST_s}{N_o}\right)$$

$$= \lim_{T_s \to 0} \frac{2ST_s}{2N_oT_s} \log_2 e = \frac{S}{N_o} \log_2 e \qquad (6.4)$$

since $\log_2(1 + x) = \log_2 e \log_e(1 + x) = \log_2 e(x - \frac{x^2}{2} + \ldots..)$.

Similarly,

$$
\begin{aligned}
\lim_{T_s \to 0} C_b &= \lim_{T_s \to 0} \frac{1}{T_s}(1 + p \log p + (1-p) \log(1-p)) \text{ bits/sec.} \\
&= \lim_{T_s \to 0} \frac{1}{T_s} \Bigg\{ 1 + \tfrac{1}{2} \left[1 - \left(\frac{2}{\pi} \right)^{\frac{1}{2}} \left(\frac{2ST_s}{N_o} \right)^{\frac{1}{2}} \right] \\
&\quad \cdot \left[\log_e \left[1 - \left(\frac{2}{\pi} \right)^{\frac{1}{2}} \left(\frac{2ST_s}{N_o} \right)^{\frac{1}{2}} \right] \log_2 e - 1 \right] \qquad (6.5) \\
&\quad + \tfrac{1}{2} \left[1 + \left(\frac{2}{\pi} \right)^{\frac{1}{2}} \left(\frac{2ST_s}{N_o} \right)^{\frac{1}{2}} \right] \left[\log_e \left[1 + \left(\frac{2}{\pi} \right)^{\frac{1}{2}} \left(\frac{2ST_s}{N_o} \right)^{\frac{1}{2}} \right] \log_2 e - 1 \right] \Bigg\} \\
&= \lim_{T_s \to 0} \frac{1}{T_s} \left\{ \tfrac{1}{2} \left(\frac{2}{\pi} \right) \frac{2ST_s}{N_o} \log_2 e \right\} = \frac{2}{\pi} \frac{S}{N_o} \log_2 e
\end{aligned}
$$

where only the first two terms of the expansions of $\log_2(1 + x)$ and the first term of $\mathrm{erf}(x) = \frac{1}{(2\pi)^{\frac{1}{2}}} \left[x - \frac{2}{3} x^3 + \ldots.. \right]$ were taken since all succeeding terms involve powers of T_s greater than the first power and may therefore be neglected. Thus, in the limit as $T_s \to 0$, the loss due to the quantization of the binary channel is represented by the factor $\frac{2}{\pi}$.

The advantages of using a binary channel are potentially great. While the orthogonal codes do asymptotically achieve channel capacity over the continuous channel, the size of the computer necessary to decode eventually becomes prohibitive. Binary codes of much greater length could be used, however, with the same amount of equipment due to the existence of more efficient decoding schemes(31). Unfortunately, if the orthogonal codes are to be used in such a way that the transmission rate, and hence the time per bit T_b is held constant, it is found that the optimum value of N is generally rather small. That is, the orthogonal code achieving the lowest error probability at a given signal-to-noise ratio is one of a relatively low order. The error probability actually becomes rapidly worse for higher order codes. Consequently, there is apparently no possibility of approaching the channel capacity with these codes over the binary symmetric channel. Figure 6.1 shows the bit error probability over the binary symmetric channel for orthogonal codes of various lengths as a function of the signal-to-noise ratio, $\frac{ST_b}{N_o}$. (The bit error probability is $\frac{N-1}{2N}$ times the word error probability; see Chapter 1.) Also included, for purposes of comparison, is the uncoded bit error probability. Note that it is always considerably more advantageous to use the channel as a continuous channel with the same codes if such a procedure is possible.

Again, analytical comparison is difficult, but the following argument gives credance to the above statement concerning the behavior of the error probability as a function of the size of the code. Consider a code that can correct up to, and including, e errors. It can be shown(32) that if $e > Np$ for all N, where N is the number of symbols per code word and p the probability of an error per symbol, then the probability of error per word asymptotically approaches zero with increasing N. However, suppose $e + 1 < Np$. The probability that a word is in error when e or fewer errors can be corrected is just:

MODEL

DATE

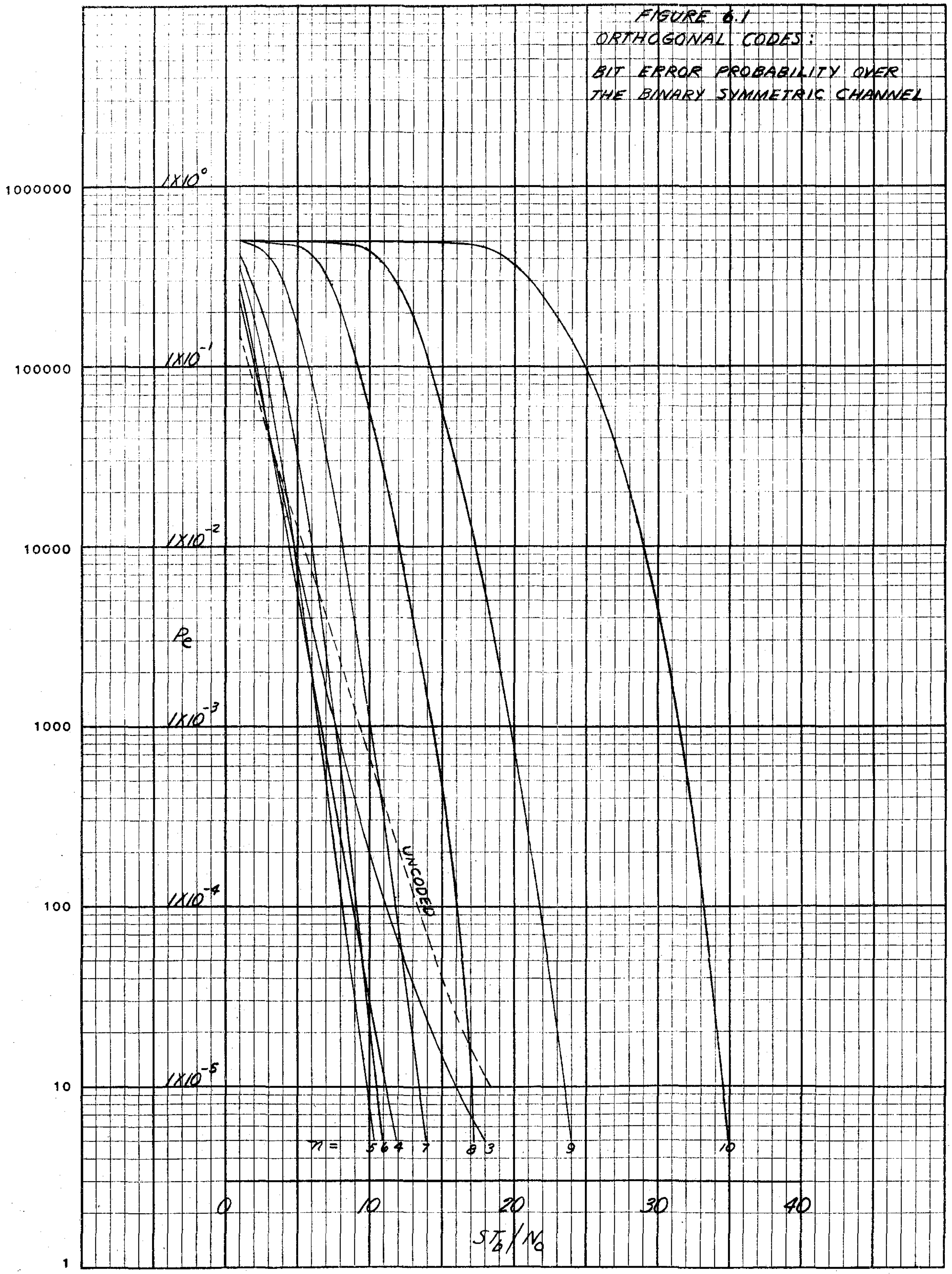


FIGURE 6.1
ORTHOGONAL CODES:

BIT ERROR PROBABILITY OVER THE BINARY SYMMETRIC CHANNEL

$$P_e = \sum_{i=e+1}^{N} \binom{N}{i} p^i(1-p)^{N-i} .$$

Let $e + 1 = Np(1 - \epsilon)$. Then by deMoivre's theorem, as N becomes large

$$P_e = \sum_{i=Np(1-\epsilon)}^{N} \binom{N}{i} p^i(1-p)^{N-i} \longrightarrow \frac{1}{(2\pi)^{\frac{1}{2}}} \int_{-N^{\frac{1}{2}}\left(\frac{p}{1-p}\right)^{\frac{1}{2}}\epsilon}^{N^{\frac{1}{2}}\left(\frac{1-p}{p}\right)^{\frac{1}{2}}} e^{-\frac{t^2}{2}} \, dt. \qquad (6.6)$$

Thus P_e increases with increasing N for all $\epsilon \geq 0$. Consequently, a reasonable figure of merit for an error correcting code is $F = \frac{e+1}{Np}$.

Now consider the orthogonal codes. If the bit rate of transmission is fixed, then for a given signal-to-noise ratio, $\frac{ST_b}{N_o}$, where T_b is the time per bit, the probability of error per symbol is:

$$p = \tfrac{1}{2} - \mathrm{erf}\left[\left(\frac{ST_b}{N_o}\right)^{\frac{1}{2}} \left(\frac{2\log_2 N}{N}\right)^{\frac{1}{2}} \right] \qquad (6.3a)$$

and e, the number of correctable errors is $\frac{N}{4} - 1$. This follows from the fact that two orthogonal words differ in exactly $\frac{N}{2}$ symbols. If fewer than $\frac{N}{4}$ errors are made in any one word, it is still closer to the original word than to any other. If $\frac{N}{4}$ or more errors are made, this is not necessarily true. Thus:

$$F = \frac{e+1}{Np} = \frac{\frac{N}{4}}{\frac{N}{2} - N \operatorname{erf}\left[\left(\frac{ST_b}{N_o} \right)^{\frac{1}{2}} \left(\frac{2\log_2 N}{N} \right)^{\frac{1}{2}} \right]} \qquad (6.7)$$

and

$$\lim_{N \longrightarrow \infty} F \longrightarrow \frac{\frac{N}{4}}{\frac{N}{2} - \frac{N^{\frac{1}{2}}}{(2\pi)^{\frac{1}{2}}} \left(\frac{ST}{N_o}\right)^{\frac{1}{2}} \left(2\log_2 N \right)^{\frac{1}{2}}} \longrightarrow \tfrac{1}{2} . \qquad (6.8)$$

But, as was shown above, if $F \leq 1$, the probability of error per word increases with increasing N and it is apparent that in this case it approaches 1 as $N \longrightarrow \infty$. It is evident that F has this same asymptotic behavior for bi-orthogonal and trans-orthogonal codes. Thus, for a fixed, finite signal-to-noise ratio, there is always an optimum, finite size for these codes. Larger codes actually increase the error probability.

It should be observed that while the number of errors correctable in an orthogonal code is $\frac{N}{4} - 1$, as many as $\frac{N - \sqrt{N}}{2}$ errors are detectable. (This corresponds to the fact that a vector may actually have a correlation of $\pm \frac{1}{\sqrt{N}}$ with every code vector, as was shown earlier.) Thus, all information is not necessarily lost if more than $\frac{N}{4} - 1$ errors are made. However, much of the advantage of algebraic decoding would be lost if this added information is to be used, since a much more detailed knowledge of the coset leaders must be obtained and exploited in this case.

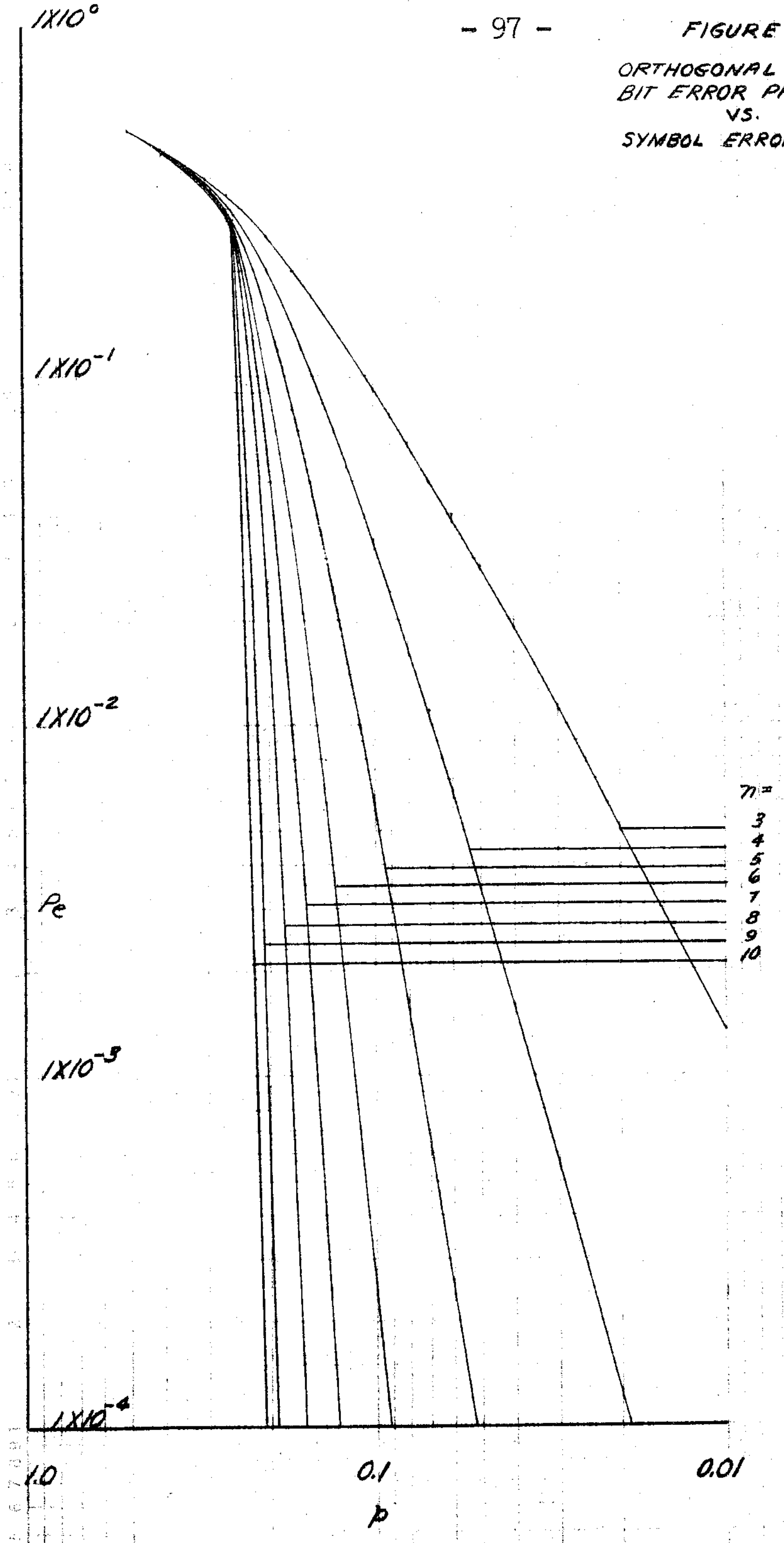
There is another commonly encountered situation in which it may be advantageous to use these codes over the binary symmetric channel. In particular, if the symbol time is constrained, orthogonal codes provide a useful means of exchanging transmission rate for a decreased error probability. While this exchange may decrease the information transmitted by the channel according to Shannon's definition (the rate of information transmission is effectively equal to the source rate since the equivocation is negligible at the error probabilities of interest), nevertheless, in many situations it is much more important to transmit one bit of information per second with a probability of error, $P_e = 10^{-5}$, for example, than to transmit ten bits with $P_e = 10^{-3}$. Figure 6.2 shows the bit error probability as a function of the symbol error probability, p, for orthogonal codes. The transmission rate is evidently reduced by the factor $\frac{n}{2^n}$.

B. Word Synchronization by the Random Source Method

Efficient decoding over the binary symmetric channel, as over the continuous channel, demands word synchronization. Techniques completely analogous to those used in earlier chapters to obtain this synchronization are applicable here, although entirely different methods are required in order to calculate the necessary convergence time. If, as in Chapter 2, it is assumed that the source is such that a completely random sequence of code words is transmitted and that each word has an equal probability of being selected, a method similar to the one discussed there can be used. Since any orthogonal group or group coset contains an equal number of ones and zeros in every symbol position but one, which is all zeros (or ones), it suffices, as before, to determine that unique position. Assume, without loss of generality, that this unique symbol is always a zero. Then this

FIGURE 6.2

ORTHOGONAL CODES:
BIT ERROR PROBABILITY
VS.
SYMBOL ERROR PROBABILITY



symbol in the received code word is a zero with probability 1-p and a one with probability p, the symbol error probability defined above. The remainder of the symbols are equally likely to be a one or a zero. The probability of correct synchronization after m words, then, is the probability that more zeros occurred in this unique symbol position than in any of the other positions. As before, a lower bound on this probability can be obtained by establishing a threshold on the number of zeros occurring at any position and determining the probability that more than this number occur at the unique position, while fewer occur at every other position. Again, the received symbols are not mutually independent due to the word structure, but this difficulty is avoided exactly as before by observing that

$$P(A_1 \cup A_2 \cup A_3 \cup \ldots \cup A_N) \leq P(A_1) + P(A_2) + \ldots + P(A_N).$$

Thus, the probability of correct synchronization, P_s, after m words and with a threshold r is bounded by:

$$P_s \geq 1 - \sum_{i=0}^{r-1} \binom{m}{i} q^i p^{m-i} - \frac{(N-1)}{2^m} \sum_{i=r}^{m} \binom{m}{i} \qquad (6.9)$$

$$= P_B(m, r, (1-p)) - (N-1) P_B(m, r, \tfrac{1}{2})$$

where $P_B(m, r, p)$ is as defined in Chapter 5. Table 6.1 shows the number of words m necessary to establish synchronization with a probability of 0.999 and of 0.9999 and the corresponding threshold for various values of the dictionary size, N. The value of p is chosen so that the synchronous word error probability is 10^{-3}.

Table 6.1

Synchronization over the Binary Symmetric Channel
Assuming a Random Source

N	p	m(.999)	r(.999)	m(.9999)	r(.9999)
8	.01	24	21	32	28
16	.03	34	29	43	36
32	.06	48	38	62	49
64	.10	72	54	84	63
128	.14	92	68	110	79

C. Word Synchronization with Comma-Free Codes

Again, the comma-free properties of certain of these codes may be used to obtain the word synchronization necessary for decoding. To take advantage of the algebraic decoding techniques which have been developed, the code to be used must be a group or a group coset. It will be observed that all the examples of comma-free codes given in Appendix A are, in fact, group cosets. Let x be the vector added to the group to render it comma-free. Then the decoding process involves subtracting (or adding, since the arithmetic is modulo two) x from the incoming word and then determining the coset, or the weight of the coset, to which the word belongs. The weight of the coset, by definition the minimum Hamming weight of any element of that coset, indicates the most probable number of errors that occurred during transmission. If this weight is less than e, the number of correctable errors, the sequence is decoded and the word that was most probably transmitted is determined.(33)

If no errors occurred during transmission the weight of the coset of the received in-phase word will be zero, while the out-of-phase weights will be at least D_o, the index of the comma freedom. (Note, the index of comma freedom will be denoted by D_o here rather than the p of previous chapters to avoid confusion with the customary use of the symbol p as the probability of error in the binary symmetric channel.) The problem is to distinguish between the two situations with the desired accuracy even though transmission errors do occur. As before, a synchronization scheme is selected with no further justification than the results that are obtained with it, and the fact that upper bounds can be calculated rigorously. The process is as follows: A threshold, D_1, is selected and the number of occurrences of elements whose cosets are of weight less than or equal to D_1 is counted at each phase position. In the in-phase case, the probability that the coset will have weight D_1 or less is given by:

$$P_o = \sum_{i=0}^{D_1} \binom{N}{i} p^i (1-p)^{N-i} . \qquad (6.10)$$

Consider now an out-of-phase sequence y with the property that $|x + y| = D$ for some code word x. The probability that a random error vector transfers y into a new vector z with the property that $|x + z| \leq D_1 < D$ can be calculated as follows: There are N-D component positions in which x and y agree, and D in which they disagree. If an error vector, which changes y to a vector agreeing with x in N-d positions, $d < D$, contains i ones in the N-D positions of agreement, then it must contain exactly D-d+i ones in the D positions of disagreement for some $i = 0, 1, 2, \ldots, d$. That is,

$$(N-D) - i + (D - d + i) \;=\; N-d.$$

The number of error vectors with this property, for a given value of i, is just $\binom{N-D}{i}\binom{D}{D-d+i}$. The probability that any one of them occurs is $p^{D-d+2i}\;q^{N-D+d-2i}$ and thus the probability that the error vector alters y in such a way that it agrees in exactly d places with x is

$$\sum_{i=0}^{d} \binom{N-D}{i}\binom{D}{D-d+i}\, p^{D-d+2i}\; q^{N-D+d-2i}.$$

The probability that $|x + z| \leq D_1$ is then

$$P\left(|x+z| \leq D_1 \,\Big|\, |x+y| = D\right) = \sum_{d=0}^{D_1} \sum_{i=0}^{d} \binom{N-D}{i}\binom{D}{D-d+i}\, p^{D-d+2i}\; q^{N-D+d-2i}$$

$$= \sum_{d=0}^{D_1} \left[\sum_{i=0}^{d} \binom{N-D}{i}\, p^{i}\, q^{N-D-i}\right]\binom{D}{D-d+i}\, p^{D-d+i}\; q^{d-i} \qquad (6.11)$$

$$= \sum_{d=0}^{D_1} \sum_{i=0}^{d} p_B(N-D,\ i,\ p)\;\; p_B(D,\ D-d+i,\ p)$$

where $p_B(n, r, p)$ is the probability of exactly r successes in n trials when the probability of a success in one trial is p.

In order to obtain an upper bound on the probability P_1 that an out-of-phase vector is mistaken for a code word, and to find one which is independent of the received error-free sequence y, it is necessary to determine the value of D such that the probability P(D) that a random error vector e causes y to fall within the threshold of a code word x or its complement $\overline{x}$ is maximized. The probability that $z = y + e$ falls within the threshold of $\overline{x}$ is given by equation 6.11 when D is replaced by N-D, or equivalently, when p and q are interchanged. Thus

$$P(D) = \sum_{d=0}^{D_1} \sum_{i=0}^{d} \left[\binom{N-D}{i}\binom{D}{d-i}\right]\left[p^{D-d+2i}\, q^{N-D+d-2i} + q^{D-d+2i}\, p^{N-D+d-2i}\right]. \tag{6.12}$$

Now consider

$$P(D) - P(D+1) =$$

$$\sum_{d=0}^{D_1} \sum_{i=0}^{d} \left[\binom{N-D-1}{i}\binom{D}{d-i} + \binom{N-D-1}{i-1}\binom{D}{d-i}\right] \cdot \left[P_{i,d} + Q_{i,d}\right] -$$

$$- \sum_{d=0}^{D_1} \sum_{i=0}^{d} \left[\binom{N-D-1}{i}\binom{D}{d-i} + \binom{N-D-1}{i}\binom{D}{d-i-1}\right]\left[\frac{p}{q}\, P_{i,d} + \frac{q}{p}\, Q_{i,d}\right]$$

where

$P_{i,d} = p^{D-d+2i} q^{N-D+d-2i}$, $Q_{i,d} = q^{D-d+2i} p^{N-D+d-2i}$, and the identity

$$\binom{n}{i} + \binom{n}{i-1} = \binom{n+1}{i}$$

was used to eliminate the terms $\binom{N-D}{i}$ and $\binom{D+1}{d-i}$. But

$$\sum_{d=0}^{D_1} \sum_{i=0}^{d} \binom{N-D-1}{i-1} \binom{D}{d-i} \left[P_{i,d} + Q_{i,d}\right]$$

$$= \sum_{d=0}^{D_1-1} \sum_{i=0}^{d} \binom{N-D-1}{i} \binom{D}{d-i} \left[\frac{p}{q} P_{i,d} + \frac{q}{p} Q_{i,d}\right]$$

where i-1 has been replaced by i, and d-1 by d. Similarly,

$$\sum_{d=0}^{D_1} \sum_{i=0}^{d} \binom{N-D-1}{i} \binom{D}{d-i-1} \left[\frac{p}{q} P_{i,d} + \frac{q}{p} Q_{i,d}\right]$$

$$= \sum_{d=0}^{D_1-1} \sum_{i=0}^{d} \binom{N-D-1}{i} \binom{D}{d-i} \left[P_{i,d} + Q_{i,d}\right]$$

where d-1 has been replaced by d. Thus,

$$P(D) - P(D+1) = (q-p) \sum_{d=0}^{D_1} \sum_{i=0}^{d} \binom{N-D-1}{i} \binom{D}{d-i} \left[\frac{1}{q} P_{i,d} - \frac{1}{p} Q_{i,d}\right]$$

$$- (q-p) \sum_{d=0}^{D_1-1} \sum_{i=0}^{d} \binom{N-D-1}{i} \binom{D}{d-i} \left[\frac{1}{q} P_{i,d} - \frac{1}{p} Q_{i,d}\right]$$

$$= (q-p) \sum_{i=0}^{D_1} \binom{N-D-1}{i} \binom{D}{D_1-i} \left[\frac{1}{q} P_{i,D_1} - \frac{1}{p} Q_{i,D_1}\right] . \qquad (6.13)$$

But this is just q-p times the probability that a vector y' with N-1 components is transformed into any of the vectors located at a distance D_1 from x' minus the probability that y' is transformed into any of those vectors at the same distance from $\overline{x}'$, where the distance from y' to x' is D and that from y' to $\overline{x}'$ is N-1-D. Now, in a method entirely analogous to that just completed, examine the expression

$$F(D') - F(D' + 1)$$

where

$$F(D') = \frac{1}{q} \sum_{i=0}^{D_1} \binom{N-D'-1}{i} \binom{D'}{D_1-i} P_i$$

and where the second subscript on P has been dropped for convenience. As before,

$$F(D') - F(D' + 1) =$$

$$\frac{1}{q} \sum_{i=0}^{D_1} \left[\binom{N-D'-2}{i} \binom{D'}{D_1-i} + \binom{N-D'-2}{i-1} \binom{D'}{D_1-i}\right] P_i$$

$$- \frac{1}{q} \sum_{i=0}^{D_1} \left[\binom{N-D'-2}{i} \binom{D'}{D_1-i} + \binom{N-D'-2}{i} \binom{D'}{D_1-i-1}\right] \frac{p}{q} P_i$$

$$= \frac{1}{q} \sum_{i=0}^{D_1} \binom{N-D'-2}{i} \binom{D'}{D_1-i} \left[P_i\left(1 - \frac{p}{q} \right)\right]$$

$$- \frac{1}{q} \sum_{i=0}^{D_1-1} \binom{N-D'-2}{i} \binom{D'}{D_1-i-1} \left[\frac{p}{q} P_i\left(1 - \frac{p}{q} \right)\right]$$

$$= \frac{(q-p)}{q^2} \binom{N-D'-2}{D_1} P_o + \sum_{i=0}^{D_1-1} \binom{N-D'-2}{i} \left[\binom{D'}{D_1-i} - \binom{D'}{D_1-i-1} \frac{p}{q} \right] P_i. \quad (6.14)$$

Although more general results may be obtained, it suffices here to observe that every term of the last summation is non-negative if $D' \geq \frac{1}{q} D_1 - 1$ and if, of course, $q > \frac{1}{2} > p$. Then if $\frac{N}{2} \geq D + 1 \geq \frac{1}{q} D_1$, $N-D-1 \geq D$, and it follows from repeated application of equation 6.14 that $F(D) - F(N-D-1)$ is positive. Consequently, equation 6.13 is positive and $P(D) > P(D + 1)$ if $\frac{N}{2} \geq D + 1 \geq \frac{1}{q} D_1$. This, in turn, states that the probability, $P(D)$, increases as D decreases, at least as long as D obeys these inequalities. But since the comma-free condition assures that $D \geq D_o$ for all received vectors y, then if $q(D_o + 1) \geq D_1$, an upper bound on the probability, P_1, that the error vector e is such that $y + e$ falls within the threshold D_1 of any code word or its complement is obtained by letting $|x + y| = D_o$ and $|\bar{x} + y| = N-D_o$ or vice versa for every code word x. Then, if $q(D_o + 1) \geq D_1$,

$$P_1 = \sum_{x \text{ and } \bar{x}} P_r(|z + x| \leq D_1) \leq$$

$$N \sum_{d=0}^{D_1} \sum_{i=0}^{d} p_B(N-D_o, i, p) \; p_B(D_o, D_o-d+i, p) \quad (6.15)$$

$$+ N \sum_{d=0}^{D_1} \sum_{i=0}^{d} p_B(D_o, i, p) \; p_B(N-D_o, N-D_o-d+i, p) \; .$$

If the sign information is already available, as is often assumed for a binary symmetric channel, the second summation may be neglected.

The probability of correct synchronization after m words, P_s, is just the probability that the received vectors fall within the threshold D_1 of a code word r or more times for the correct phase position and fewer than r times for every other phase position, for some predetermined value of r. Hence, as before, P_s is bounded by

$$P_s \geq P_B(m, r, P_o) - (N-1)\, P_B(m, r, P_1).$$

$P_B(m, r, p)$ is, as before, the cumulative binomial distribution:

$$P_B(m, r, p) = \sum_{i=r}^{m} \binom{m}{i} p^i (1-p)^{m-i} \ .$$

Table 6.2 lists the number of words necessary to establish synchronization, with the probability .999 and .9999 for various values of dictionary size, N. The threshold D_1 is also included, as well as the values of P_o and P_1 of equations 6.10 and 6.12, respectively. Note that $(1-p)(D_o + 1)$ is, indeed, greater than D_1 in every case, and hence that the condition, upon which the above argument concerning the upper bound on P_1 depends, is satisfied.

Table 6.2

Synchronization with Comma-Free Codes over the Binary Symmetric Channel

N	p	D_o	D_1	P_o	P_1	m(.999)	r(.999)	m(.9999)	r(.9999)
16	.03	2	0	.614	.009	16	4	21	5
32	.06	6	2	.677	.0016	15	3	19	3
64	.10	14	6	.539	$6.62\mathrm{x}10^{-6}$	13	2	16	2
128	.14	34	20	.827	$2.56\mathrm{x}10^{-8}$	4	1	6	1

An interesting estimate of the expected synchronization time can be obtained under the assumption that the overlaps, including the errors, constitute a uniform random distribution over the possible 2^N binary vectors of N components. That is, any given vector has probability 2^{-N} of occurring as a particular overlap. Again, picking a threshold D_1, the sequence corresponding to the correct phase position will be within distance D_1 of a word with probability P_o,

$$P_o \geq \sum_{i=0}^{D_1} \binom{N}{i} p^i q^{N-i} \qquad (6.10)$$

where, as before, the possibility that a word might be altered in such a way that it is within a distance D_1 of some other word has been neglected. The probability that a particular out-of-phase sequence is within D_1 of a code word is given by:

$$P_1 = \frac{N}{2^N} \sum_{i=0}^{D_1} \binom{N}{i} \qquad (6.17)$$

since there are exactly $N \sum_{i=0}^{D_1} \binom{N}{i}$ different vectors of N components within this distance of a code word. The probability of synchronization after m word times is, as before,

$$P_s \gtrsim P_B(m, r, P_o) - (N-1)\ P_B(m, r, P_1) . \tag{6.16}$$

The value of m necessary for $P_s = .999$ and $P_s = .9999$ is shown in Table 6.3.

Table 6.3

Synchronization Assuming A Random Distribution of Out-of-Phase Vectors

N	p	D_1	m(.999)	r(.999)	m(.9999)	r(.9999)	P_o	P_1
8	.01	1	10	9	15	13	.999	.28
16	.03	3	8	7	10	9	.999	.17
32	.06	7	5	4	6	5	.999	.034
64	.10	15	3	2	4	3	.999	$\sim 10^{-3}$
128	.14	31	~ 1	1	2	1	.999	$\sim 10^{-6}$

The results in Table 6.2, it should be noted, are rigorous upper bounds on the number of words necessary for synchronization, regardless of the sequence of words transmitted. The <u>expected</u> number of words given in Table 6.3, however, were calculated under the unjustified assumption that the out-of-phase vectors constitute a random distribution in which any one of the 2^N binary vectors of N components is equally likely. A heuristic justification of this assumption can be obtained by the same device as that employed in Chapter 4. That is, if a random vector is added to every element of an orthogonal group to form the code dictionary, then any sequence formed by an overlap has random properties. The error vector is certainly random and consequently any given out-of-phase vector is random-like. To the extent that there tends to be a correspondance between codes with this randomness property and comma-free codes, the results in Table 6.3 are not unreasonable for these codes. The fact that, for codes of length 128 and greater, essentially only one word may be necessary for synchronization is at first disconcerting. This suggests that probability of correct detection for long codes is effectively the same with or without synchronization. However, this is merely a reflection of the fact that the longer codes become progressively less efficient as error-correcting codes. That is, the $N \sum_{i=0}^{\frac{N}{4}-1} \binom{N}{i}$ decodable vectors become a vanishingly small part of the 2^N possible vectors. If the out-of-phase vectors are truly random, the probability that any one of them can be decoded is negligible. It should be emphasized that this discussion refers only to the expected synchronization time. The comma-free properties of these same codes, however, assure that the upper bounds given previously are valid regardless of the randomness, or lack of randomness, of the received sequence.

Chapter 7

A METHOD OF ANALYZING THE COMMA-FREE PROPERTIES OF GROUP CODES

A. The Matrix Equation

It was shown in Chapter 3 that all orthogonal groups of order N are equivalent, under a permutation of columns, to the group generated by the elements:

```
00.........0011.........11
00....01....1100....011...11
.                        .
.                        .
.                        .
0101..................0101   .
```

This n by $2^n = N$ matrix will be referred to as the generator matrix G. Note that no two column vectors of G are equivalent. This may be verified as follows: The first row establishes that none of the first $\frac{N}{2}$ columns can be identical to any of the second $\frac{N}{2}$; the second that none of the first $\frac{N}{4}$ columns can equal any of the second $\frac{N}{4}$ and similarly for the third and fourth quarters. The first and second rows then guarantee that if any column is repeated, the second must occur in the same quarter of the matrix as the first. It readily follows that the third row provides that any two identical columns must occur in the same $\frac{1}{2^3}$rd $= \frac{1}{8}$th of the matrix, etc., until finally the nth row divides the matrix into 2^n sections, no two of which can contain identical columns. Thus, since all 2^n columns must be distinct binary vectors of n components, all of the possible 2^n vectors are represented. As a result, any binary orthogonal group generator may be represented by a matrix whose columns are some permutation of the 2^n binary numbers of n digits. Similarly, any element in this orthogonal group may be represented by the expression $y = H^T x$ where H^T is the transpose of a matrix H obtained by some permutation of the columns of G, x is any of the 2^n binary vectors of n components and all arithmetic operations are over the binary field, GF(2).(34) This simply represents the fact that any member of the group may be obtained from some linear combination of the generators. Any group coset may be obtained by adding a fixed 2^n-tuple to the corresponding group elements. Thus any group coset element may be expressed as

$$y = H^T x + c \tag{7.1}$$

where c is some 2^n-tuple, designated, for convenience, the coset leader.

Now let $H_f^T(k)$ be the k x n matrix consisting of the first k rows of H^T and, similarly, let $H_e^T(k)$ comprise the last k rows of H^T. In addition, let c_k be a vector formed by cyclically permuting the coset leader in such a way that if δ_i is the ith component of $c = c_o$, then the ith component of c_k is δ_{i+k} where the subscript is to be interpreted modulo N. Then any sequence of length $N = 2^n$ formed by the last k digits of a code word of the above coset followed by the first N - k digits of another (not necessarily different) code word is readily seen to be expressible as

$$\begin{bmatrix} H_e^T(N - k) \\ \\ O(k) \end{bmatrix} y + \begin{bmatrix} O(N - k) \\ \\ H_f^T(k) \end{bmatrix} z + c_k$$

where $O(j)$ is an $j \times n$ matrix of zeros, and y and z are $n \times 1$ binary vectors. Thus the sum of any overlap with any code word or its complement may be represented by:

$$w\,I + H^T x + c_o + \begin{bmatrix} H_e^T(N - k) \\ \\ O(k) \end{bmatrix} y + \begin{bmatrix} O(N - k) \\ \\ H_f^T(k) \end{bmatrix} z + c_k \tag{7.2}$$

$$= \begin{bmatrix} H_f^T(N - k) & H_e^T(N - k) & O(N - k) & I_{(N - k)} \\ \\ H_e^T(k) & O(k) & H_f^T(k) & I(k) \end{bmatrix} \begin{bmatrix} x \\ y \\ \\ z \\ w \end{bmatrix} + c_o + c_k$$

$$\equiv M_k u + c_o + c_k$$

where w is a binary scalar (a one component vector), and $I(k)$ is a $k \times 1$ vector consisting of all ones. If w is 0, the overlap is compared to a code word; if 1, the comparison is with the complement of a code word. Note that M is an $N \times 3n + 1$ matrix and u a $3n + 1$ component vector. Finally, the index of comma freedom is given by

$$\min_{u,k} \left| M_k u + c_o + c_k \right| ,$$

that is, by the minimum Hamming weight of the above vector.

Let e be the vector of minimum weight such that

$$M_{k_o} u_o + c_o + c_{k_o} = e$$

for some $u = u_o$ and for some $k = k_o$. Then

$$M_{k_o} u_o = c_o + c_{k_o} + e$$

since addition and subtraction are equivalent over GF(2). For this to be true, a well-known theorem of matrix algebra[35] states that

$$\rho(M_{k_o}) = \rho(M_{k_o}, c_o + c_{k_o} + e) \tag{7.3}$$

where ρ denotes "rank", and $M_{k_o}, c + c_{k_o} + e$ is the matrix M_{k_o} augmented by the column vector $c + c_{k_o} + e$.* Thus, for example, it is easy to establish whether a given coset is a comma-free code by letting e be the all zeros vector and determining whether $\rho(M_k) = \rho(M_k, c + c_k)$ for any value of k. $\rho(A)$ is perhaps most readily determined by reducing A to Hermite canonical form[36] since the necessary operations are trivial over GF(2).

* Note that all arithmetic operations here are over the field GF(2). Most of the results of matrix algebra, however, and certainly all those techniques which are used here, are valid for any field.

Unfortunately, no such convenient method exists for determining the minimum value of $|e|$ when it is not zero. The following approach is useful for determining an upper bound on $|e|$ for a given group, for evaluating $|e|$ min for a particular coset, and in some cases as a method of constructing an optimum coset leader for a given group. The calculation involved in the above three investigations, however, becomes increasingly tedious as the code length increases, and the first of these is probably the only technique which is useful for any but the smallest orthogonal codes.

The procedure is as follows: An operator matrix A is determined so that:

$$AM_k = \binom{B}{0} \qquad (7.4)$$

where B is an $r \times 3n + 1$ matrix such that if the first one in row j occurs in the kth column, then the first one in row j + 1 occurs in the k + mth column where $m \geq 1$, and 0 is an $N-r \times N-r$ matrix of zeros. This is the Hermite canonical form. The rank of M_k is obviously r. If P_{ij} is a matrix which permutes the ith and jth rows, and E_{ij} a matrix which adds the ith to the jth row, then

$$A = P_{i_1j_1} E_{i_2j_2} P_{i_3j_3} \cdots\cdots E_{i_nj_n}$$

where the elementary matrices are determined in the step-by-step reduction of M_k. Now applying A to the above matrix equation, one obtains:

$$AMu = A(c + c_k) + A\ e.$$

Since $\rho(M_k) = r$, the vector, u, can always be chosen so that the first r of the above equations can be solved. The left side of the remaining N-r equations is always zero, and hence e must be chosen so that

$$\widetilde{A(c + c_k)} = \widetilde{A\,e} \tag{7.5}$$

where the "tilde" indicates the truncated vectors consisting of the last N-r components only. There are 2^N ways of selecting the vector e, but only 2^{N-r} possible values for $\widetilde{A\,e}$. Thus, if, beginning with the vectors of lowest weight, the 2^{N-r} vectors for e are selected such that every possible $\widetilde{A\,e}$ has been obtained, then the vector of maximum weight necessary to complete this set represents the maximum possible index of comma freedom. This index can be obtained if a vector c which satisfies the above equation exists, and if, for every other value of k, the vector c also necessitates a vector e of at least this minimum weight. While this method could conceivably be used to construct an optimum coset leader, the process is evidently extremely laborious. However, it has been found that the upper bounds obtained by investigating a given group in the above manner for a single value of k, in particular, $k = \frac{N}{4} + 1$ or $\frac{3N}{4} - 1$, are generally attainable. This technique then is valuable in eliminating much unnecessary search for a better coset leader. In addition, if a coset leader has already been selected, this method facilitates the investigation of the index of comma freedom since it is given by the minimum weight of e necessary to satisfy the above equation. Several examples of the application of this method to obtain least upper bounds are given in Appendix B.

The above technique is somewhat simplified in the investigation of one very important orthogonal group, the cyclic group. A cyclic group code is defined here to be a code in which every word is a cyclic shift of a particular sequence. Except for the case $N = 4$, no strictly cyclic orthogonal codes are known. However, cyclic trans-orthogonal codes exist for all values of $N = 2^n - 1$(37) and orthogonal codes can easily be constructed from these by adjoining a zero to the beginning of every word. The advantage of such a dictionary is that it can be generated by an extremely small amount of equipment. The property of these codes that is useful here is that the generator matrix H^T obeys a linear recurrence relationship.(38) That is, each row of H^T can be obtained by taking the sum of no more than $n = \log_2 N$ previous rows.

$$h_k = \sum_{i=1}^{n} a_i h_{k-i} \qquad \text{for all } k,$$

where a_i is an element of the field GF(2), and "row sum" designates the sums of the corresponding components, also over GF(2). The matrix A is thus obtained most readily in this case, as is illustrated in Appendix B.

It should also be observed that the techniques of this chapter are equally applicable to the bi-orthogonal case. By substituting the two columns $\begin{matrix} O(N-k) & I(N-k) \\ I(k) & O(k) \end{matrix}$ for the single column $I(N)$ in the matrix $M(k)$ and adding the additional variable w' to the vector u, it is possible to complement the first and second parts of an overlap independently. This is the only change necessary to investigate a bi-orthogonal code for comma freedom (cf. Appendix A).

Note that, since the expected weight of e increases as the number of different vectors $\overset{\frown}{A}$ e increases, this weight will tend to be greater for larger values of 2^{N-r} and hence for minimum rank r. The code generated by the matrix G itself is particularly attractive in this regard since, as may be readily verified, cyclic permutations of the elements of G are very frequently also in G or in the complement of G. This indicates that the matrix M_n constructed from G, contains a large number of columns which are linear combinations of other columns, and hence that the rank of M is generally considerably less than $3n + 1$.

B. <u>The Non-Existence of Orthogonal Comma-Free Group Codes of Eight Words</u>

Consider the binary orthogonal dictionaries consisting of eight words. The matrix M_2 and M_3 may be represented, for an arbitrary dictionary, by:

$$M_2 = \begin{array}{cccc} \alpha_1 & \alpha_3 & \emptyset & 1 \\ \alpha_2 & \alpha_4 & \emptyset & 1 \\ \alpha_3 & \alpha_5 & \emptyset & 1 \\ \alpha_4 & \alpha_6 & \emptyset & 1 \\ \alpha_5 & \alpha_7 & \emptyset & 1 \\ \alpha_6 & \alpha_8 & \emptyset & 1 \\ \alpha_7 & \emptyset & \alpha_1 & 1 \\ \alpha_8 & \emptyset & \alpha_2 & 1 \end{array} \qquad M_3 = \begin{array}{cccc} \alpha_1 & \alpha_4 & \emptyset & 1 \\ \alpha_2 & \alpha_5 & \emptyset & 1 \\ \alpha_3 & \alpha_6 & \emptyset & 1 \\ \alpha_4 & \alpha_7 & \emptyset & 1 \\ \alpha_5 & \alpha_8 & \emptyset & 1 \\ \alpha_6 & \emptyset & \alpha_1 & 1 \\ \alpha_7 & \emptyset & \alpha_2 & 1 \\ \alpha_8 & \emptyset & \alpha_3 & 1 \end{array} \qquad (7.5)$$

where α_i is some member of the field, $GF(2^3)$, represented as one of the eight binary numbers of 3 digits and $\emptyset$ is the three-tuple, 000. If a comma-free code is to exist, then the rank of both M_2 and M_3 must be less than eight since $\rho(M_k, c_o + c_k)$ cannot be greater than eight. Thus there must be some linear combination over GF(2) of the rows of each of these matrices which is equal to the vector of all zeros.

More precisely, it may be stated that some linear combination of exactly four rows of M_2 and M_3 must be identically zero for the following reasons: (1) The last column of M_2 and M_3 contains all ones, and hence no odd number of rows can sum to zero, (2) No combination of two rows can be identically zero since this would imply that $\alpha_i = \alpha_j$ for $i \neq j$, (3) It is easily verified that $\sum_{i=1}^{8} \alpha_i = \emptyset \equiv (000)$ and hence, if some linear combination of six rows of M_2 or M_3 is equal to zero, there are two α_i's whose sum is $\emptyset$, again implying $\alpha_i = \alpha_j$, $i \neq j$, and (4) If the only linear combination which yields the zero vector involves all eight rows, then $\rho(M) = \rho(M, c_o + c_k)$, and no comma freedom is possible in this case, either. This is because the sum of all of the components $\gamma_i + \gamma_{i+k}$ of the vector $c_o + c_k$ is necessarily also zero; that is,

$$\sum_{i=1}^{8} (\gamma_i + \gamma_{i+k}) = \sum_{i=1}^{8} \gamma_i + \sum_{i=1}^{8} \gamma_i = 0.$$

It will be assumed that $\alpha_1, \alpha_2, \alpha_3 \neq \emptyset$. If this is not true, the following arguments may instead be applied to M_5 and M_6. Consider M_3. Number these rows $s_i (i = 1, 2, \ldots, 8)$. If $\alpha_1 + \alpha_2 + \alpha_3 \neq \emptyset$, four of the first five rows must sum to zero if the rank is to be less than eight. Suppose s_1, s_2, s_3 and s_4 are linearly dependent. Then $\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = \emptyset$, $\alpha_4 + \alpha_5 + \alpha_6 + \alpha_7 = \emptyset$, and hence $\alpha_1 + \alpha_2 + \alpha_3 + \alpha_5 + \alpha_6 +$

$\alpha_7 = \emptyset = \alpha_4 + \alpha_8$ which implies $\alpha_4 = \alpha_8$. Since this is impossible, these four rows cannot be linearly dependent. But any four of the first five rows of M_3 contain either α_4 or α_5 twice. If only one of the two is contained twice, a linear combination of six α_i's may be equated to $\emptyset$ implying, as before, that $\alpha_h = \alpha_e$, $h \neq e$. Thus the only possibility left is that both α_4 and α_5 are involved twice in the linear combination, and hence that s_1, s_2, s_4 and s_5 are linearly dependent.

If $\alpha_1 + \alpha_2 + \alpha_3 = \emptyset$ then s_6, s_7, s_8, s_i $(i = 1, 2, \ldots, 5)$ may be linearly dependent. Evidently i corresponds to that α_{i+3} which is equal to $\emptyset$. If $i = 5$, $\alpha_8 = \emptyset$ and $\alpha_5 + \alpha_6 + \alpha_7 = \emptyset = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_5 + \alpha_6 + \alpha_7$ which implies $\alpha_4 = \alpha_8$. Similar conflicts result if $i = 4$, 2 or 1. The only possibility remaining is that $i = 3$ and $\alpha_6 = \emptyset$. Hence if the rank of M_3 is to be less than eight (except in the uninteresting case (4) above) either $\alpha_1 + \alpha_2 + \alpha_4 + \alpha_5 = \alpha_4 + \alpha_5 + \alpha_7 + \alpha_8 = \emptyset$ or $\alpha_1 + \alpha_2 + \alpha_3 = \alpha_6 = \alpha_3 + \alpha_6 + \alpha_7 + \alpha_8 = \emptyset$. But since $\alpha_1 + \alpha_2 + \alpha_3 + \alpha_6 = \emptyset$ implies that $\alpha_4 + \alpha_5 + \alpha_7 + \alpha_8 = \emptyset$ and $\alpha_3 + \alpha_6 + \alpha_7 + \alpha_8 = \emptyset$ implies that $\alpha_1 + \alpha_2 + \alpha_4 + \alpha_5 = \emptyset$ the second situation implies the first.

Now consider M_2 and identify its rows by r_i $(i = 1, 2, \ldots, 8)$. Since $\alpha_1 \neq \alpha_2$, four of the first six rows of M_2 must be linearly dependent, if the rank is to be less than eight (again excluding case (4)). The first column in the following table lists all $\binom{6}{4} = 15$ such possibilities and the second column indicates the conflict, if any, resulting if $\alpha_1 + \alpha_2 + \alpha_4 + \alpha_5 = \emptyset = \alpha_4 + \alpha_5 + \alpha_7 + \alpha_8$.

Table 7.1

Linear Dependent Rows	Implication if $\alpha_1 + \alpha_2 + \alpha_4 + \alpha_5 = \emptyset$
r_1 r_2 r_3 r_4	$\alpha_3 = \alpha_5$
r_1 r_2 r_3 r_5	$\alpha_3 = \alpha_4$
r_1 r_2 r_3 r_6	$\alpha_3 = \alpha_7$
r_1 r_2 r_4 r_5	$\alpha_4 = \alpha_8$
r_1 r_2 r_4 r_6	$\alpha_4 = \alpha_7$
r_1 r_2 r_5 r_6	$\alpha_4 = \alpha_6$
r_1 r_3 r_4 r_5	$\alpha_2 = \alpha_3$
r_1 r_3 r_4 r_6	$\alpha_2 = \alpha_4$
r_1 r_3 r_5 r_6	$\alpha_2 = \alpha_5$
r_1 r_4 r_5 r_6	$\alpha_2 = \alpha_6$
r_2 r_3 r_4 r_5	$\alpha_1 = \alpha_3$
r_2 r_3 r_4 r_6	$\alpha_1 = \alpha_4$
r_2 r_3 r_5 r_6	$\alpha_1 = \alpha_5$
r_2 r_4 r_5 r_6	$\alpha_1 = \alpha_6$
r_3 r_4 r_5 r_6	$\alpha_3 = \alpha_5$

Since the condition that $\alpha_1 + \alpha_2 + \alpha_4 + \alpha_5 = \emptyset = \alpha_4 + \alpha_5 + \alpha_7 + \alpha_8$ makes it impossible for any four of the rows of M_2 to be linearly dependent, there are no comma-free orthogonal group codes and, a fortiori, no comma-free bi-orthogonal group codes of order eight. It has been shown(39) that all orthogonal, and hence all bi-orthogonal, codes containing 12 or fewer words are in the same equivalence class. Since the above method includes investigation of all dictionaries in the equivalence class of the bi-orthogonal group codes, it follows that there are no bi-orthogonal comma-free codes consisting of eight words, which is as contended in Chapter 4.

Chapter 8

SUPPLEMENTARY REMARKS CONCERNING SYNCHRONIZATION

A. Other Approaches to Synchronization

Before concluding this paper, it would be well to mention some approaches others have taken in attempting to overcome the synchronization problem. Most of the effort in this area has been orientated toward the binary symmetric channel and has been concerned with uncoded data transmission. The work of Gilbert(40) has already been mentioned. This paper investigates the maximum number, G(N), of N symbol code words which have the following property: Each code word is to have the same A symbol prefix, and all other blocks of A consecutive symbols occurring in an arbitrary sequence of code words are to be different from this prefix. Neither the possibility of error-correction nor the cross-correlations of the resulting code words is investigated, and no estimate is made concerning the time necessary for synchronization.

Another approach results in the binary N-tuples known as the Barker sequences(41). These are sequences, $\left\{ x_i \right\}$, which have the property that their aperiodic autocorrelation:

$$a(k) = \sum_{i=1}^{N-k} (1 - 2x_i)(1 - 2x_{i+k}) = \begin{cases} N & k = 0 \\ \pm 1, 0 & k \neq 0 \end{cases} \qquad (8.1)$$

Such sequences are known to exist for N = 1, 2, 3, 4, 5, 7, 11, 13. They are known not to exist for odd N > 13. Sequences with $|a(k)| \leq n$, $k \neq 0$, for larger integers n have been found for many lengths, N. These and related sequences can be used in the synchronization of the uncoded binary symmetric channel in a straightforward manner. If such an N-tuple is inserted periodically in a random sequence of binary digits, one N-tuple for every M data bits, then the reception of this pattern tends to indicate synchronization.(42) There are three sources of errors in the synchronization process: (1) The synchronizing pattern can occur in the random part of the sequence consisting of data bits, (2) The pattern can occur as a combination of m random bits preceeded, or followed, by N-m bits of the true pattern, and (3) The true pattern may be missed due to the random errors. The purpose in using Barker-like sequences is, of course, to decrease the probability of an error due to (2).

An alternative scheme to accomplish the same task is simply to send a synchronization bit every D data bits.(43,44) If this bit is always zero (or always one) while the data bits are equally likely to be zero or one, then the time necessary for synchronization is calculated essentially as in Chapter 6.

It is interesting to note that these two techniques for synchronization are in many essentials equivalent. In particular, if one synchronizing bit is sent after every D data bits in the second scheme, the synchronization time is seen to be virtually the same as that of the first if the ratio of M to N is D. The effect due the type (2) error in the first process is certainly negligible for moderately large values of D. The second method has the advantage that synchronization information is received after every D bits, as opposed to every M bits in the first method. This advantage may well be counterbalanced, however, if the frame consists of more than D bits. A frame is a block of data digits which carry all the results of a series of experiments or observations. Frame synchronization is often necessary in order that the user know to which observation a particular set of data bits pertains. Thus, if one synchronizing bit per frame is not enough, it may be necessary to transmit additional frame synchronization information when using the second method.

As was mentioned, the random source synchronization procedure described in Chapter 6 is essentially the same as the methods described above, except that the data were assumed to be uncoded in the latter. As error-correcting codes, in fact, the orthogonal codes offer no advantage over the trans-orthogonal codes since the column consisting of all zeros (or all ones) can be neither an information bit nor a non-trivial parity check bit. In this context, then, the constant column of the orthogonal code may be considered as a device added to the trans-orthogonal codes only for the purpose of synchronization. The same statement may be made concerning these codes as used in Chapter 2, except that, as observed earlier, the negative cross-correlation of the trans-orthogonal codes may sometimes be disadvantageous. Under these circumstances, the added column is useful both in the synchronization and in the decoding mode of operation. To the extent that this column has no use other than synchronization, however, these codes violate condition (2) in the Introduction. Nevertheless, as was shown in Chapter 1, the effective power, for the continuous channel, is decreased by at most a factor of $\frac{N-1}{N}$, which is negligible for all but the shortest codes. The synchronizing methods discussed in this section, of course, all suffer from the same defect to a greater or lesser degree, depending upon the ratio of the number of synchronizing bits to data bits. These methods are in addition subject to the previously mentioned difficulties presented by the highly probable non-random source. All are significantly less efficient that the comma-free codes of moderately large index, even when the source does appear to be random.

B. Frame Synchronization

No mention has been made in previous chapters concerning frame synchronization. If this information is necessary, however, it is particularly simple to obtain with orthogonal codes. The procedure is to send the code word in complemented form to designate the beginning of a new frame. To evaluate the effect of this perturbation on the operation of the channel, observe that the increase in the probability of a word error over the continuous channel due to the possibility of a complemented word is negligible. This may be verified by comparing the orthogonal and bi-orthogonal word error probabilities given in Chapter 1. The difference is insignificant for $N \geq 16$. The probability of missing the frame synchronization is then essentially equivalent to that of making a word error in synchronous operation.

The difference in word synchronization time caused by the infrequent occurrence of a word complement is also negligible, at least for the comma-free codes. It will be noticed in Appendix A that the maximum index of comma freedom found is, in every case, the same for the orthogonal and bi-orthogonal codes. Thus, even if the complemented form of the word occurred with high frequency, the synchronization time would be unchanged. In the random source technique of synchronization in Chapter 2 it is easily verified that the expected value of the integral I_o is decreased by a factor of $\frac{M-2}{M}$ is a frame consists of M words. Similarly, the variance of I_o is increased from $\sigma_n^2 \frac{T}{N}$ to $\sigma_n^2 \frac{T}{N} + \frac{4A^2T^2}{N^2} \frac{M-1}{M^2}$. The effective signal-to-noise ratio is thus reduced from

$$\frac{A^2T}{\sigma^2 N} = R\left(\frac{2\log_2 N}{N} \right) = R_{SN}$$

to

$$R\left(\frac{2\log_2 N}{N} \right) \left[\frac{\left(\frac{M-2}{M} \right)^2}{1 + 4R\left(\frac{2\log_2 N}{N} \right) \left(\frac{M-1}{M^2} \right)} \right]$$

$$\approx R\left(\frac{2\log_2 N}{N}\right) \left[1 - \frac{4}{M} R\left(\frac{2\log_2 N}{N} \right) \right] = R_{SN} - \frac{4}{M} R_{SN}^2 .$$

Since R is generally less than five, this perturbation is seen to be small for large values of M. The mean and variance of $I_{k\neq 0}$ is obviously unchanged by this process.

Similar comments apply to the synchronous error probabilities, and to the word synchronization time when this frame synchronization method is used over the binary symmetric channel. For the same reason as before almost nothing is changed by this procedure when the comma-free properties of these codes are used. When the synchronizing symbol is used, the probability of a <u>one</u> occurring in this symbol position is evidently changed from p to $\frac{p(M-1) + (1-p)}{M} = p + \frac{1-2p}{M}$ which may or may not be significant depending upon the relationship between M and p. Note that the synchronizing symbol need not be complemented, in this case, with the rest of the frame word. This leaves the word synchronization time unaltered, but does alter the word error probability. This is because the complement of a trans-orthogonal code word is "closer" to the other dictionary words than the word itself. However, it can be verified that this change is also slight. Which of these two methods is superior would depend on several factors, particularly the values of M and p.

Note that the disadvantage of using a bi-orthogonal code (i.e., the lack of sign information) is not a consideration here even though the sign of a word is given significance. The great imbalance between the number of frame synchronization words ("negative" sign) and the remainder of words ("positive" sign) enables immediate distinction to be made between the two cases.

In summary, the frame synchronization problem can be solved with virtually no alteration in the results obtained in the previous chapters when comma-free codes are used. The random source synchronization techniques are slowed up somewhat by this perturbation but, particularly for the continuous channel, these effects tend to be slight. This method has a special advantage over that mentioned in the previous section in that the probability of an error in obtaining the frame synchronization in one attempt is extremely low and consequently the frames need not be equal in length.

C. Retention of Word Synchronization

It should be observed, for the sake of completeness, that it is possible to lose word synchronization after it has been obtained and that the optimum procedure for the retention of word synchronization is not necessarily that used to obtain it initially. In particular, the most probable way for synchronization to be lost is by a shift of one symbol in either direction from the correct position. This type of error is easily guarded against by occasionally, or continuously, monitoring the positions on either side of the position believed to be correct. This process also results in continually increased confidence in the correctness of the accepted phase position when word synchronization has, indeed, been obtained. The criterion for deciding that the word synchronization is in error is not essentially different than the methods discussed above and hence will not be investigated further here.

Appendix A

EXAMPLES OF COMMA-FREE DICTIONARIES

Various orthogonal and bi-orthogonal dictionaries have been investigated for comma freedom on an IBM-7090 digital computer, and some of the codes having a high index of comma freedom are presented here. The computer programs for dictionaries of words of 16 and 32 binary symbols were essentially equivalent and completely straightforward in nature. Sequences were formed by taking N (N = 16 or 32) consecutive binary digits from an overlap formed by adjoining two words. All $N^2(N-1)$ such sequences, in the orthogonal case, were compared to all N code words, and the maximum and minimum number of agreements was recorded. For the bi-orthogonal case, another $N^2(N-1)$ sequences formed by taking N symbols from an overlap of one code word followed by the complement of another were investigated. It is evident that no other sequence need be considered, since those formed from the overlap of the complement of one word followed by a second word, or from the complements of two words, are just the complements of sequences already investigated. Since the number of agreements between two N-tuples

$$|x + y| = w$$

implies

$$|x + \overline{y}| = N-w$$

all of the necessary information is available after consideration of the two sets of sequences mentioned above.

Note that since there are approximately N^4 comparisons to be made, the computation time should increase by a factor of 16 as the code length doubles. This, in fact, was what happened in going from N = 16 to N = 32.

However, since the computer word length is 36 bits, complications result when codes with word lengths greater than 36 are to be investigated. The computation time necessary for N = 64, while still not exorbitant, was about sixty times as great as that for N = 32. At this rate of increase, the N = 128 case was impossibly long. Thus for this dictionary size a somewhat more restricted program was used which took even less time than that necessary for N = 64. The program was restricted in the sense that it considered only code dictionaries of the form:

$$\begin{matrix} A & A & A & A \\ A & A & \bar{A} & \bar{A} \\ A & \bar{A} & A & \bar{A} \\ A & \bar{A} & \bar{A} & A \end{matrix}$$

where A is a 32 x 32 orthogonal array and $\bar{A}$ is its complement. Any binary vector of length N = 128 could be added (modulo two) to each of the above code words. Thus, investigation of the large dictionary (N = 128) for comma freedom could be done essentially by considering only the smaller (N = 32) dictionaries independently and combining the results.

Examples of bi-orthogonal codes of various lengths are presented below. In every case, it was found that the best index of comma freedom obtainable with an orthogonal code could also be obtained with a bi-orthogonal code of the same word length. Only the orthogonal half of these codes is presented; the remainder of the code dictionary consists of the complements of the words shown. N, of course, designates the number of symbols in the code words, while p denotes the index of comma freedom associated with them. All of these examples are of the form G of Chapter 7 with a coset leader c which is some phase shift of a pseudo-random sequence of length 2^n-1 plus an additional symbol. These coset leaders are identified in each dictionary with an asterisk and the extra symbol is underlined.

Example 1

$N = 16$, $p = 2$

*0111$\underline{0}$10101100100
0111010110011011
0111101001101011
0111101010010100
0100011001010111
0100011010101000
0100100101011000
0100100110100111
0010000000110001
0010000011001110
0010111100111110
0010111111000001
0001001100000010
0001001111111101
0001110000001101
0001110011110010

Example 2

N = 32, p = 6

```
*10001101110101000010010110011111
 10001101110101001101101001100000
 10001101001010110010010101100000
 10001101001010111101101010011111
 10000010110110110010101010010000
 10000010110110111101010101101111
 10000010001001000010101001101111
 10000010001001001101010110010000
 10111110111001110001011010101100
 10111110111001111110100101010011
 10111110000110000001011001010011
 10111110000110001110100110101100
 10110001111010000001100110100011
 10110001111010001110011001011100
 10110001000101110001100101011100
 10110001000101111110011010100011
 11011000100000010111000011001010
 11011000100000011000111100110101
 11011000011111100111000000110101
 11011000011111101000111111001010
 11010111100011100111111111000101
 11010111100011101000000000111010
 11010111011100010111111100111010
 11010111011100011000000011000101
 11101011101100100100000111111001
 11101011101100101011111000000110
 11101011010011010100001100000110
 11101011010011011011110011111001
 11100100101111010100110011110110
 11100100101111011011001100001001
 11100100010000100100110000001001
 11100100010000101011001111110110
```

Example 3

N = 64, p = 14

```
0001110001101001000100111101010000000011000100001100010111000010
0001110010010110000100110010101100000011111011111100010100111101
0001001101100110000111001101101100001100000111111100101011001101
0001001110011001000111000010010000001100111000001100101000110010
0010111101011010001000001110011100110000001000111111011011110001
0010111110100101001000000001100000110000110111001111011000001110
0010000001010101001011111110100000111111001011001111100111111110
0010000001010101000101111000101110011111111010011111110010000001
0100100100111100010001101000000101010110010001011001000010010111
0100100111000011010001100111111001010110101110101001000001101000
0100011000110011010010011000111001011001010010101001111110011000
0100011011001100010010010111000101011001101101011001111101100111
0111101000001111011101011011001001100101011101101010001110100100
0111101011110000011101010100110101100101100010011010001101011011
0111010100000000011110101011110101101010011110011110110010101011
0111010111111111011110100100001001101010100001101010110001010100
0001110001101001000100111101010011111100111011110011101000111101
0001110010010110000100110010101111111110000010000001110101100010
0001001101100110000111001101101111100111111000000011010100110010
0001001110011001000111000010010011110011000111110011010111001101
0010111101011010001000001110011111001111110111000000100100001110
0010111110100101001000000001100011001111001000110000100111110001
0010000001010101001011111110100011000000110100110000011000000001
0010000001010101000101111000101111100000000101100000001101111110
0111101000001111011101011011001010011010100010010101110001011011
0111101011110000011101010100110110011010011101100101110010100100
0111010100000000011110101011110110010101100001100101001101010100
0111010111111111011110100100001010010101011110010101001110101011
0100100100111100010001101000000110101001101110100110111101101000
0100100111000011010001100111111010101001010001010110111110010111
0100011000110011010010011000111010100110101101010110000001100111
0100011011001100010010010111000110100110010010100110000010011000
0001110001101001111011000010101100000011000100000011101000111101
0001110010010110111011001101010000000011111011110011101011000010
0001001101100110111000110010010000001100000111110011010100110010
0001001110011001111000111101101100001100111000000011010111001101
0010111101011010110111110001100000110000001000110000100100001110
0010111110100101110111111110011100110000110111000000100111110001
0010000001010101110100000001011100111111001011000000011000000001
0010000001010101011010000111010000011111111010011000001101111110
0100100100111100101110010111111001010110010001010110111101101000
0100100111000011101110011000000101010110101110100110111110010111
0100011000110011101101100111000101011001010010100110000001100111
0100011011001100101101101000111001011001101101010110000010011000
0111101000001111100010100100110101100101011101100101110001011011
0111101011110000100010101011001001100101100010010101110010100100
```

Example 3 (continued)

N = 64, p = 14

```
 0111010100000000100001010100001001101010011110010101001101010100
 0111010111111111100001011011110101101010100001100101001110101011
 0001110001101001111011000010101111111100111011111110001011100010
*0001110010010110111011001101010011111110000010000110001010011101
 0001001101100110111000110010010011110011111000001100101011001101
 0001001110011001111000111101101111110011000111111100101000110010
 0010111101011010110111110001100011001111110111001111011011110001
 0010111110100101110111111110011111001111001000111111011000001110
 0010000000101010111010000000101111000000110100111111100111111110
 0010000001010101011010000111010001100000001011001111100100000001
 0100100100111100101110010111111010101001101110101001000010010111
 0100100111000011101110011000000110101001010001011001000001101000
 0100011000110011101101100111000110100110101101011001111110011000
 0100011011001100101101101000111010100110010010101001111101100111
 0111101000001111100010100100110110011010100010011010001110100100
 0111101011110000100010101011001010011010011101101010001101011011
 0111010100000000100001010100001010010101100001101010110010101011
 0111010110111111100001011011110110010101011110011010110001010100
```

Example 4

N = 128, p = 34

The dictionary matrix is of the form

$$\begin{array}{cccc} A & B & C & D \\ A & B & \overline{C} & \overline{D} \\ A & \overline{B} & C & \overline{D} \\ A & \overline{B} & \overline{C} & D \end{array}$$

where

Example 4 (continued)

```
A =

     *11111111000000100000110000101000
      11111111000000101111001111010111
      11111111111111010000110011010111
      11111111111111011111001100101000
      11110000000011010000001100100111
      11110000000011011111110011011000
      11110000111100100000001111011000
      11110000111100101111110000100111
      11001100001100010011111100011011
      11001100001100011100000011100100
      11001100110011100011111111100100
      11001100110011101100000000011011
      11000011001111100011000000010100
      11000011001111101100111111101011
      11000011110000010011000011101011
      11000011110000011100111100010100
      10101010010101110101100101111101
      10101010010101111010011010000010
      10101010101010000101100110000010
      10101010101010001010011001111101
      10100101010110000101011001110010
      10100101010110001010100110001101
      10100101101001110101011010001101
      10100101101001111010100101110010
      10011001011001000110101001001110
      10011001011001001001010110110001
      10011001100110110110101010110001
      10011001100110111001010101001110
      10010110011010110110010101000001
      10010110011010111001101010111110
      10010110100101000110010110111110
      10010110100101001001101001000001
```

Example 4(continued)

B =

```
*11110010001011001110101001111101
 11110010001011000001010110000010
 11110010110100111110101010000010
 11110010110100110001010101111101
 11111101001000111110010101110010
 11111101001000110001101010001101
 11111101110111001110010110001101
 11111101110111000001101001110010
 11000001000111111101100101001110
 11000001000111110010011010110001
 11000001111000001101100110110001
 11000001111000000010011001001110
 11001110000100001101011001000001
 11001110000100000010100110111110
 11001110111011111101011010111110
 11001110111011110010100101000001
 10100111011110011011111100101000
 10100111011110010100000011010111
 10100111100001101011111111010111
 10100111100001100100000000101000
 10101000011101101011000000100111
 10101000011101100100111111011000
 10101000100010011011000011011000
 10101000100010010100111100100111
 10010100010110101000110000011011
 10010100010110100111001111100100
 10010100101001011000110011100100
 10010100101001010111001100011011
 10011011010101011000001100010100
 10011011010101010111110011101011
 10011011101010101000001111101011
 10011011101010100111110000010100
```

Example 4 (continued)

C =

```
*00001110001001001101101011011110
 00001110001001000010010100100001
 00001110110110111101101000100001
 00001110110110110010010111011110
 00000001001010111110101011101000 1
 00000001001010110010101000101110
 00000001110101001101010100101110
 00000001110101000010101011010001
 00111101000101111110100111101101
 00111101000101110001011000010010
 00111101111010001110100100010010
 00111101111010000001011011101101
 00110010000110001110011011100010
 00110010000110000001100100011101
 00110010111001111110011000011101
 00110010111001110001100111100010
 01011011011100011000111110001011
 01011011011100010111000001110100
 01011011100011101000111101110100
 01011011100011100111000010001011
 01010100011111101000000010000100
 01010100011111100111111101111011
 01010100100000011000000001111011
 01010100100000010111111110000100
 01101000010000101011110010111000
 01101000010000100100001101000111
 01101000101111011011110001000111
 01101000101111010100001110111000
 01100111010011011011001110110111
 01100111010011010100110001001000
 01100111101100101011001101001000
 01100111101100100100110010110111
```

Example 4 (continued)

D =

```
*11000110100101110111001100101010
 11000110100101111000110011010101
 11000110011010000111001111010101
 11000110011010001000110000101010
 11001001100110000111110000100101
 11001001100110001000001111011010
 11001001011001110111110011011010
 11001001011001111000001100100101
 11110101101001000100000000011001
 11110101101001001011111111100110
 11110101010110110100000011100110
 11110101010110111011111100011001
 11111010101010110100111100010110
 11111010101010111011000011101001
 11111010010101000100111111101001
 11111010010101001011000000010110
 10010011110000100010011001111111
 10010011110000101101100110000000
 10010011001111010010011010000000
 10010011001111011101100101111111
 10011100110011010010100101110000
 10011100110011011101011010001111
 10011100001100100010100110001111
 10011100001100101101011001110000
 10100000111100010001010101001100
 10100000111100011110101010110011
 10100000000011100001010110110011
 101000000000111011110101001001100
 10101111111111100001101001000011
 10101111111111101110010110111100
 10101111000000010001101010111100
 10101111000000011110010101000011
```

Appendix B

SOME PARTICULAR UPPER BOUNDS ON THE INDEX OF COMMA FREEDOM USING THE MATRIX EQUATION

Consider the matrix

$$M_{11} = \left\{ \begin{array}{cccc} G_f^T(5) & G_e^T(5) & 0(5) & I(5) \\ G_e^T(11) & 0(11) & G_f^T(11) & I(11) \end{array} \right\}$$

where G^T is a 16 x 4 matrix of the type described in Chapter 7. Substituting for G^T, one obtains

```
0000110100001
1000001100001
0100101100001
1100011100001
0010111100001
1010000000001
0110000010001
1110000001001
0001000011001
1001000000101
0101000010101
1101000001101
0011000011101
1011000000011
0111000010011
1111000001011
```

It may be verified that the matrix, A, necessary to reduce M_{11} to the desired form is:

$$\begin{bmatrix}
 & 1 & & & & & & & & & & & & & & \\
 & & 1 & & & & & & & & & & & & & \\
 & & & & 1 & & & & & & & & & & & \\
 & & & & & & & & 1 & & & & & & & \\
1 & & & & & & & & & & & & & & & \\
 & & 1 & & 1 & & 1 & & & & & & & & & \\
1 & 1 & 1 & 1 & & & & & & & & & & & & \\
1 & 1 & & & 1 & 1 & & & & & & & & & & \\
 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & & & & & & & & \\
 & & & & & & 1 & 1 & 1 & 1 & & & & & & \\
 & 1 & 1 & 1 & 1 & & 1 & 1 & 1 & & & & & 1 & & \\
 & & & & & 1 & 1 & & & 1 & 1 & & & & & \\
 & & & & & 1 & & 1 & & 1 & & 1 & & & & \\
 & 1 & 1 & 1 & 1 & & 1 & 1 & & 1 & & & 1 & & & \\
 & & & & & & 1 & 1 & & & & & & & 1 & 1 \\
 & & & & & 1 & & 1 & & & & & & 1 & & 1
\end{bmatrix}$$

where the zeros have been omitted for clarity. The last five rows of AM_{11} are seen to contain all zeros and, hence, the rank of M_{11} is 11. Consider the vector $v = \tilde{A}e$ as defined in Chapter 7. Since there are 2^5 distinct vectors v, and since $\binom{16}{0} + \binom{16}{1} = 17 < 2^5$, the vector e must have weight at least two for some of the vectors v. It remains to be determined whether any vector v necessitates a vector e of weight greater than two. Thus only vectors v of weight three or greater need be considered. Let e_i be the <u>ith</u> component of e. It is easily verified that the following vectors $v = \tilde{A}e$ can be generated by vectors e with the indicated components one and the rest zero:

v	value of i for which $e_i = 1$
00111	6, 10
01011	12, 14
10011	6, 12
01101	8
10101	7, 14
11001	6, 15
01110	8, 14
10110	7
11010	6, 16
11100	10
01111	8, 15
10111	7, 16
11011	6
11101	10, 16
11110	10, 15

Thus all vectors v can be generated by vectors e of weight two or less, and the maximum index of comma freedom for cosets of this orthogonal group is two. Several orthogonal and bi-orthogonal cosets of this group have been found which do achieve this maximum index and have been included in Appendix A.

By way of comparison a second generator matrix H was formed by selecting random ordering of the binary numbers from 0 to 15. The resulting matrix M_{11}

$$\begin{array}{c}
0001101100001\\
0011011100001\\
0100100100001\\
0010010100001\\
0000110100001\\
0110000000011\\
1010000000111\\
1100000001001\\
1110000000101\\
1111000000001\\
1000000001101\\
1011000010101\\
0111000011001\\
1001000011101\\
0101000011111\\
1101000010001
\end{array}$$

can be reduced to the desired form by the matrix:

$$A = \begin{bmatrix}
&&&&&&&&&&1&&&&& \\
&&1&&&&&&&&&&&&& \\
&&1&1&&&&&&&&&&&& \\
1&&&&&&&&&&&&&&& \\
1&1&&1&&&&&&&&&&&& \\
1&1&&1&1&&&&&&&&&&& \\
&1&&1&&&&&&&1&&&1&& \\
&&1&1&1&&&&1&&1&&&&& \\
&&&&&&&&1&1&1&&&1&& \\
1&1&1&1&&&&&&&&&&1&&1 \\
1&1&1&1&&&&1&&&1&&&&& \\
1&&1&&1&1&&&&&&&&&& \\
1&&1&&&&&1&&1&1&&1&1&& \\
1&1&&&1&1&1&&1&&&&&&& \\
1&&&1&&&&&1&&1&1&&1&& \\
1&&&1&1&1&&1&&1&&&1&1&1&1
\end{bmatrix}$$

The last three rows of the matrix AM_{11} contain only zeros. The rank of AM_{11} and hence M is thus 13, the maximum possible value. It is easily verified, by repeating the techniques of the last example, that the maximum index of comma freedom for a coset of this orthogonal group is one.

Finally consider the generator matrix, H. whose columns are formed by the following recursion formula:

$$\alpha_i = \alpha_{i-3} + \alpha_{i-4}$$

where $\alpha_1 = 1000$, $\alpha_2 = 0100$, $\alpha_3 = 0010$ and $\alpha_4 = 0001$. The reader may verify that all fifteen non-zero binary 4-tuples are generated in this manner. Let the first column of H be the 4-tuple $0000 \equiv \emptyset$. Then:

$$M_{11} = \begin{bmatrix} \emptyset & \alpha_{123} & \emptyset & 1 \\ \alpha_1 & \alpha_{234} & \emptyset & 1 \\ \alpha_2 & \alpha_{1234} & \emptyset & 1 \\ \alpha_3 & \alpha_{134} & \emptyset & 1 \\ \alpha_4 & \alpha_{14} & \emptyset & 1 \\ \alpha_{12} & \emptyset & \emptyset & 1 \\ \alpha_{23} & \emptyset & \alpha_1 & 1 \\ \alpha_{34} & \emptyset & \alpha_2 & 1 \\ \alpha_{124} & \emptyset & \alpha_3 & 1 \\ \alpha_{13} & \emptyset & \alpha_4 & 1 \\ \alpha_{24} & \emptyset & \alpha_{12} & 1 \\ \alpha_{123} & \emptyset & \alpha_{23} & 1 \\ \alpha_{234} & \emptyset & \alpha_{34} & 1 \\ \alpha_{1234} & \emptyset & \alpha_{124} & 1 \\ \alpha_{134} & \emptyset & \alpha_{13} & 1 \\ \alpha_{14} & \emptyset & \alpha_{24} & 1 \end{bmatrix}$$

where $\alpha_{ijk} = \alpha_i + \alpha_j + \alpha_k$. By adding linear combinations of the second, third, fourth and fifth rows to the remaining rows, the matrix may be reduced in an obvious manner to the following form:

$$M = \begin{bmatrix}
\emptyset & \alpha_{123} & \emptyset & 1 \\
\alpha_1 & \alpha_{234} & \emptyset & 1 \\
\alpha_2 & \alpha_{1234} & \emptyset & 1 \\
\alpha_3 & \alpha_{134} & \emptyset & 1 \\
\alpha_4 & \alpha_{14} & \emptyset & 1 \\
\emptyset & \alpha_1 & \emptyset & 1 \\
\emptyset & \alpha_2 & \alpha_1 & 1 \\
\emptyset & \alpha_3 & \alpha_2 & 1 \\
\emptyset & \alpha_4 & \alpha_3 & 0 \\
\emptyset & \alpha_{12} & \alpha_4 & 1 \\
\emptyset & \alpha_{23} & \alpha_{12} & 1 \\
\emptyset & \alpha_{34} & \alpha_{23} & 0 \\
\emptyset & \alpha_{124} & \alpha_{34} & 0 \\
\emptyset & \alpha_{13} & \alpha_{124} & 1 \\
\emptyset & \alpha_{24} & \alpha_{13} & 0 \\
\emptyset & \alpha_{123} & \alpha_{24} & 1
\end{bmatrix}$$

Similarly:

$$A_2A_1M_{11} = \begin{bmatrix} \emptyset & \emptyset & \alpha_{12} & 0 \\ \alpha_1 & \alpha_{234} & \emptyset & 1 \\ \alpha_2 & \alpha_{1234} & \emptyset & 1 \\ \alpha_3 & \alpha_{134} & \emptyset & 1 \\ \alpha_4 & \alpha_{14} & \emptyset & 1 \\ \emptyset & \alpha_1 & \emptyset & 1 \\ \emptyset & \alpha_2 & \alpha_1 & 1 \\ \emptyset & \alpha_3 & \alpha_2 & 1 \\ \emptyset & \alpha_4 & \alpha_3 & 0 \\ \emptyset & \emptyset & \alpha_{14} & 1 \\ \emptyset & \emptyset & \emptyset & 1 \\ \emptyset & \emptyset & \emptyset & 1 \\ \emptyset & \emptyset & \alpha_{14} & 0 \\ \emptyset & \emptyset & \alpha_{14} & 1 \\ \emptyset & \emptyset & \emptyset & 1 \\ \emptyset & \emptyset & \alpha_{14} & 0 \end{bmatrix}$$

$$A_3A_2A_1M_{11} = \begin{bmatrix} \emptyset & \emptyset & \alpha_{24} & 1 \\ \alpha_1 & \alpha_{234} & \emptyset & 1 \\ \alpha_2 & \alpha_{1234} & \emptyset & 1 \\ \alpha_3 & \alpha_{134} & \emptyset & 1 \\ \alpha_4 & \alpha_{14} & \emptyset & 1 \\ \emptyset & \alpha_1 & \emptyset & 1 \\ \emptyset & \alpha_2 & \alpha_1 & 1 \\ \emptyset & \alpha_3 & \alpha_2 & 1 \\ \emptyset & \alpha_4 & \alpha_3 & 0 \\ \emptyset & \emptyset & \emptyset & 1 \\ \emptyset & \emptyset & \emptyset & 1 \\ \emptyset & \emptyset & \emptyset & 1 \\ \emptyset & \emptyset & \alpha_{14} & 0 \\ \emptyset & \emptyset & \emptyset & 1 \\ \emptyset & \emptyset & \emptyset & 1 \\ \emptyset & \emptyset & \emptyset & 0 \end{bmatrix}$$

and finally:

$$A_4A_3A_2A_1M_{11}=PAM_{11}=\begin{bmatrix} \emptyset & \emptyset & \alpha_{24} & 1 \\ \alpha_1 & \alpha_{234} & \emptyset & 1 \\ \alpha_2 & \alpha_{1234} & \emptyset & 1 \\ \alpha_3 & \alpha_{134} & \emptyset & 1 \\ \alpha_4 & \alpha_{14} & \emptyset & 1 \\ \emptyset & \alpha_1 & \emptyset & 1 \\ \emptyset & \alpha_2 & \alpha_1 & 1 \\ \emptyset & \alpha_3 & \alpha_2 & 1 \\ \emptyset & \alpha_4 & \alpha_3 & 0 \\ \emptyset & \emptyset & \emptyset & 0 \\ \emptyset & \emptyset & \emptyset & 0 \\ \emptyset & \emptyset & \emptyset & 0 \\ \emptyset & \emptyset & \alpha_{14} & 0 \\ \emptyset & \emptyset & \emptyset & 1 \\ \emptyset & \emptyset & \emptyset & 0 \\ \emptyset & \emptyset & \emptyset & 0 \end{bmatrix}$$

Thus the matrix A is very easily obtained in this instance. In the example here, M_{11} evidently has rank 11, and, as in the first example, the maximum index of comma freedom attainable may be shown to be two.

References

1. Claude E. Shannon and Warren Weaver, The Mathematical Theory of Communication, (1959), pp. 64-73.

2. P. M. Woodward, Probability and Information Theory with Applications to Radar, (1955), pp. 62-80.

3. P. M. Woodward and I. L. Davis, Proc. Inst. Elect. Engrs., (1952), Vol. 99, Part III, pp. 37-45.

4. R. M. Fano, Communication Theory, (1953), pp. 169-182.

5. George L. Turin, IRE Trans. on Information Theory, (1960), Vol. IT-6, No. 3, pp. 311-329.

6. R. M. Fano, Transmission of Information, (1961), pp. 148-157.

7. E. N. Gilbert, IRE Trans. on Information Theory, (1960), Vol. IT-6, No. 4, pp. 470-476.

8. Shannon, op. cit., pp. 49-63.

9. Wilbur B. Davenport and William L. Root, An Introduction to the Theory of Random Signals and Noise, (1958), pp. 154-157.

10. A. J. Viterbi, IRE Trans. on Space Electronics and Telemetry, (1961), Vol. SET-7, No. 1, pp. 3-14.

11. Davenport and Root, op. cit., pp. 154-155.

12. Harold Cramer, Mathematical Methods of Statistics, (1945), pp. 370-378.

13. E. J. Gumbel, Statistics of Extremes, (1958), pp. 126-143.

14. Nakibe T. Uzgoren, *Studies in Mathematics and Mechanics*, (1954), pp. 346-353.

15. Viterbi, *op*. *cit*., p. 10.

16. Fano, *op*. *cit*., p. 157.

17. Shannon, *op*. *cit*., p. 67.

18. Albert H. Nuttall, *Technical Report TR-61-1-BF*, Litton Systems, Inc., (1961), pp. 7-59.

19. A. V. Balakrishnan, *Journal of Math. Analysis and Applications*, (1961), pp. 346-366.

20. *Ibid*., p. 350.

21. L. Baumert, M. Easterling, S. Golomb and A. Viterbi, *Technical Report No. 32-67*, Jet Propulsion Laboratory, (1961), pp. 4-5.

22. R. Turyn, *Final Report F475-1*, Sylvania Electronic Systems, (1960), p. IV-6.

23. Viterbi, *op*. *cit*., pp. 5-12.

24. Nuttall, *op*. *cit*., pp. 38-59.

25. W. W. Peterson, *Error-Correcting Codes*, (1961), pp. 52-54.

26. S. W. Golomb, *Terminal Progress Report*, Glenn L. Martin Company, (1955), pp. 15-20.

27. Computation Laboratory Staff, Harvard University, *Tables of the Cumulative Binomial Probability Distribution*, (1955).

28. Computation Laboratory Staff, Harvard University, *Tables of the Error Function and Its First Twenty Derivatives*, (1952).

29. E. C. Posner, To be submitted to the *Journal of the Soc. of Ind. and App. Math.*

30. Shannon, *op. cit.*, pp. 34-42.

31. Peterson, *op. cit.*, pp. 201-216.

32. R. M. Fano, *Transmission of Information*, (1961), pp. 231-240.

33. Peterson, *op. cit.*, pp. 30-41.

34. A. A. Albert, *Fundamental Concepts of Higher Algebra*, (1956), pp. 223-233.

35. S. Perlis, *Theory of Matrices*, (1952), p. 45.

36. *Ibid.*, pp. 49-52.

37. Golomb, *op. cit.*, pp. 5-22.

38. *Ibid.*, pp. 6-8.

39. Marshall Hall, *JPL Research Summary 36-10*, (1961), Vol. I.

40. Gilbert, *op. cit.*, pp. 470-476.

41. Turyn, *op. cit.*, p. I-1 to I-4.

42. E. R. Hill and J. L. Weblemoe, *Proc. of the National Telemetering Conference*, (1961), pp. 11-87 to 11-106.

43. G. Goode and J. Phillips, *Proc. of the National Telemetering*

Conference, (1961), pp. 11-15 to 11-50.

44. M. W. Williard, Proc. of the National Telemetering Conference, (1961), pp. 11-51 to 11-74.